

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені ІГОРЯ СІКОРСЬКОГО»

Теплоенергетичний факультет

Кафедра автоматизації проектування енергетичних процесів і систем

До захисту допущено

Завідувач кафедри

О.В. Коваль

(підпис)

(ініціали, прізвище)

“ ” 2019р.

**ДИПЛОМНА РОБОТА**  
**на здобуття ступеня бакалавра**

з напрямку підготовки 6.050101 “Комп’ютерні науки”

на тему “Модуль генерації гідроакустичного сигналу в плоско-  
паралельному хвилеводі променевим методом”

Виконав (-ла): студент (-ка) 4 курсу, групи ТР-51

Дудко Андрій Володимирович

(прізвище, ім’я, по батькові)

(підпис)

Керівник Старший викладач, к.т.н, Варава І.А.

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

(підпис)

Рецензент

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

(підпис)

Засвідчую, що у цій дипломній роботі немає  
запозичень з праць інших авторів без  
відповідних посилань.

Студент

(підпис)

Київ – 2019

**Національний технічний університет України  
“Київський політехнічний інститут імені Ігоря Сікорського”**

Факультет теплоенергетичний

Кафедра автоматизації проектування енергетичних процесів і систем

Рівень вищої освіти перший, бакалаврський

Напрямок підготовки 6.050101 “Комп’ютерні науки”

ЗАТВЕРДЖУЮ

Завідувач кафедри

\_\_\_\_\_ О.В. Коваль  
(підпис)

” \_\_\_\_ ” \_\_\_\_\_ 2019р.

**ЗАВДАННЯ**

**на дипломну роботу студенту**

Дудко Андрій Володимирович

(прізвище, ім’я, по батькові)

1. Тема роботи \_\_\_\_\_ “Модуль генерації гідроакустичного сигналу в плоско-паралельному хвилеводі променевим методом”

керівник роботи \_\_\_\_\_ Варава Іван Андрійович, старший викладач, к.т.н.  
(прізвище, ім’я, по батькові науковий ступінь, вчене звання)

затверджена наказом вищого навчального закладу від ” \_\_\_\_ ” \_\_\_\_ 201\_\_р. № \_\_\_\_

2. Строк подання студентом роботи \_\_\_\_\_

3. Вихідні дані до роботи \_\_\_\_\_ Мова програмування C# у середовищі VisualStudio 2017 у вигляді вебсервісу за технологією ASP.NET MVC

4. Зміст розрахунково-пояснювальної записки (перелік завдань, які потрібно розробити) \_\_\_\_\_ Ознайомитися з теоретичною частиною моделювання сигналів, дослідити існуючі програмні засоби для моделювання сигналів. Розробити алгоритм та реалізувати програмний продукт для генерації гідроакустичного сигналу в плоско-паралельному хвилеводі променевим методом.

5. Перелік ілюстративного матеріалу

\_\_\_\_\_ «Актуальність розробки», «Мета та завдання роботи», «Однорідний океан постійної глибини», «Променеве представлення», «Коливальна швидкість», «Розроблений додаток», «Висновки».

7. Дата видачі завдання ” \_\_\_\_ ” \_\_\_\_\_ 201\_\_р.

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітки
1.	Вивчення та аналіз задачі	01.12.2018-31.03.2019	
2.	Розробка архітектури та загальної структури системи	01-18.04.2019	
3.	Розробка структур окремих підсистем	19-25.04.2019	
4.	Програмна реалізація системи	26.04-13.05.2019	
5.	Оформлення пояснювальної записки	06.05-01.06.2019	
6.	Захист програмного продукту	25.05.2019	
7.	Передзахист	01.06.2019	
8.	Захист	18.06.2019	

Студент

\_\_\_\_\_

(підпис)

\_\_\_\_\_

(прізвище та ініціали,)

Дудко А.В.

Керівник роботи

\_\_\_\_\_

(підпис)

\_\_\_\_\_

(прізвище та ініціали,)

Варава І.А.

## АНОТАЦІЯ

Метою дипломної роботи є створення програмного продукту для генерації гідроакустичного сигналу в плоско-паралельному хвилеводі променевим методом.

Об'єктом дослідження є способи та алгоритми моделювання сигналів. Було виконано огляд існуючих програмних застосунків для моделювання сигналів та ознайомитися із проблемами моделювання гідроакустичних сигналів, розроблено програмний продукт генерації гідроакустичних сигналів, який реалізовано методом уявних джерел для розрахунку поля тиску в плоско-паралельному хвилеводі, даний метод відноситься до променевих моделей.

Створена програмний продукт може бути використаний, як частина системи для моделювання гідроакустичних об'єктів та для наукових досліджень.

Загальний обсяг роботи: 67 сторінок, 19 ілюстрацій, 17 бібліографічних посилань та 3 додатки.

Ключові слова: гідроакустичний сигнал, моделювання гідроакустичних сигналів, метод уявних джерел, променевий метод, модуль генерації сигналів.

## **ABSTRACT**

The purpose of the thesis is to create a program product for generating a hydroacoustic signal in a plane-parallel waveguide beam method.

The objects of research are the methods and algorithms of signal simulation. An overview of the existing software applications for simulation of signals and the problems of modeling of hydroacoustic signals was performed, the program software of generation hydroacoustic signals, implemented by the imaginary sources for calculating the field of pressure in a plane-parallel waveguide, was implemented, this method belongs to beam models.

The created program product can be used as part of the system for modeling hydroacoustic objects and for scientific research.

Total volume of work: 67 pages, 19 illustrations, 17 bibliographic references and 3 attachments.

Keywords: hydroacoustic signal, modeling of hydroacoustic signals, method of imaginary sources, beam method, module of signal generation.

# ЗМІСТ

Перелік умовних позначень, скорочень і термінів .....	7
Вступ .....	8
1 Задача розробки модулю генерації гідроакустичного сигналу в плоско-паралельному хвилеводі променевим методом .....	9
2 Опис предметної області .....	13
2.1 Гідроакустика .....	13
2.2 Технології гідроакустики .....	14
2.3 Програмні продукти для моделювання сигналів Actran та Adams .....	16
3 Моделювання гідроакустичного сигналу .....	22
3.1 Математичні моделі хвильових процесів .....	22
3.2 Перехід до двовимірних координат .....	23
3.3 Однорідний океан постійної глибини .....	25
3.4 Променеве представлення .....	26
3.5 Розрахунок коливальної швидкості .....	28
3.6 Альтернативний метод моделювання сигналу .....	31
4 Засоби програмної реалізації .....	38
4.1 Середовище Visual Studio 2017 .....	38
4.2 Платформа .NET Core .....	40
5 Опис програмної реалізації модуля генерації гідроакустичного сигналу променевим методом .....	45
Висновки .....	50
Список використаних джерел .....	51
Додаток 1 .....	53
Додаток 2 .....	55
Додаток 3 .....	63

## **ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ**

1. ГАС — Гідроакустична система.
2. ГІС — геоінформаційна система; поєднує карту території та різні дані, в основному табличного типу.
3. MSC — MacNeal-Schwendler Corporation (Компанія розробник, що спеціалізується в програмних продуктах для моделювання).
4. CDT — це інструмент океанографії, який використовується для вимірювання провідності, температури і тиску морської води (D означає "глибину", яка тісно пов'язана з тиском. CDT — conductivity, temperature, and pressure).

## ВСТУП

В реаліях сьогодення в кожній галузі людської діяльності використовуються програмні комплекси та системи для досягнення певних цілей. Гідроакустика знайшла широке застосування в морській навігації, війсьній сфері, океанологічних дослідженнях, виявлення об'єктів, неоднорідностей морського середовища та ін. В останні роки у галузі гідроакустики активно розвивається напрямок пов'язаний з моделюванням акустичних сигналів. Спочатку цей напрямок почав використовуватись для розпізнавання об'єктів у військових цілях, однак останнім часом посилився інтерес до розробки гідроакустичних систем, зокрема до моделювання гідроакустичних сигналів.

Генерація гідроакустичного сигналу морського об'єкта є однією з фундаментальних задач гідроакустичних систем. Розв'язок цієї задачі може бути використаний безпосередньо або як складова частина більш об'ємної системи.

Однією з головних особливостей підводних звуків є мале згасання, тому під водою звуки можуть поширюватися на більші відстані, ніж у повітрі [1].

Враховуючи особливості акваторію (такі як глибина, кут нахилу, температура та інші) актуальною є задача автоматизації досліджень моделювання гідроакустичних сигналів. Променевий метод представлення акустичного поля сигналу дає можливість Реалізацією одного з методів моделювання сигналу є розроблений програмний продукт.



# **1 ЗАДАЧА РОЗРОБКИ МОДУЛЮ ГЕНЕРАЦІЇ ГІДРОАКУСТИЧНОГО СИГНАЛУ В ПЛОСКО- ПАРАЛЕЛЬНОМУ ХВИЛЕВОДІ ПРОМЕНЕВИМ МЕТОДОМ**

Сигналами в гідроакустичних інформаційних системах величини акустичного поля (хвилі), що змінюються у часі та просторі і відображають сигнал сукупністю своїх параметрів. У загальному випадку говорять про просторово-часових сигналах, фізичним носієм яких є векторне акустичне поле, оскільки рух частинок середовища описується вектором швидкості коливань. Для цього необхідно обчислити векторне поле коливальних швидкостей і скалярний поле тисків.

Важливим і складним в гідроакустиці є складання математичних співвідношень, що зв'язують характеристики об'єктів і корисних сигналів, тобто розробка відповідних моделей сигналів, вид яких визначається фізичною суттю процесу, ступенем його пізнання і метою дослідження. У загальному випадку опис сигнального поля являє собою функцію часових і просторових координат. Після просторової обробки в приймальні антени або якщо приймальна антена має малі розміри в порівнянні з довжиною хвилі сприйманого коливання в скалярному полі тисків, сигнал являє собою лише функцію часу.

При розробці математичних моделей враховується ряд суперечливих обмежень: з одного боку модель повинна бути адекватна реальним об'єктам, з іншого боку вона повинна бути простою, зручною для застосування.

Моделлю гідроакустичного процесу є його математичне уявлення, яке дозволяє обчислити (або постулювати) імовірнісні характеристики процесу, які цікавлять дослідника в конкретному завданні.

Вдосконалення математичного, алгоритмічного та програмного забезпечення сучасних гідроакустичних та гідролокаційних систем неможливо уявити без

моделювання роботи їх частин на попередніх випробуваннях. Повне тестове покриття в даному випадку багато в чому залежить від правильності математичної моделі в гідроакустичних системах з врахуванням обчислювальної продуктивності системи.

Саме обмеження обчислювальної продуктивності системи комплексів моделювання часто не дає можливості прямого використання математичних моделей в режимі реального часу, тому виникає необхідність розробки окремих програмних модулів для обрахунку необхідних частин, таких як гідроакустичні сигнали.

Розроблений програмний продукт розраховує компоненти гідроакустичного сигналу, та формує із них записаний в файл сигнал. Програмний продукт генерує сигнали, що мають загальний вигляд, але умови їх моделювання можуть бути легко розширені та адаптовані відносно виконання поставленої задачі.

Метою роботи було створення системи для автоматизації процесу розрахунків та генерації гідроакустичного сигналу з заданими параметрами:

- координати джерела сигналу;
- координати Гідроакустичної системи(ГАС);
- частоту випромінювання джерела сигналу;
- час моделювання;
- глибина моря.

Моделюваний сигнал обмежений границями хвилеводу та граничним кутом відбиття для верхньої та нижньої границь. Для моделювання використовується променевий метод, який є ефективним для ближньої зони. Ближня зона — це ділянка акустичного поля, де тиск звуку варіюється від мінімуму до максимуму, в даному випадку приблизно на відстані в 300 метрів від джерела до приймача.

Робота системи закінчується отриманням кінцевого dat-файлу в якому міститься чотири канали:  $V_x$ ,  $V_y$ ,  $V_z$ ,  $P$ . Це значення коливальних швидкостей та сумарного акустичного тиску в певний момент часу моделювання, вони і будуть характеризувати наш сигнал для його запису в файл.

Змодельований сигнал може бути легко перевірений за допомогою частотного спектра, який є амплітудо-частотною характеристикою сигналу. На графіку спектра в своєму піку та можливо в гармоніках, які можна отримати при розкладі сигналу до ряду Фур'є, повинна виділятися задана при моделюванні частота.

Розроблена система повинна забезпечувати наступні можливості:

- моделювання гідроакустичного сигналу;
- отримання dat-файлу.

Програмний продукт може бути використаний в багатьох сферах гідроакустики, зокрема як:

- як частина системи моделювання гідроакустичних процесів, об'єктів та систем(рисунок 1);
- в системах порівняння сигналів;
- в дослідженні різних видів сигналів;
- для перевірки коректності роботи частин гідролокаційних систем;
- для перевірки алгоритмів та пристроїв виділення над шумом сигналів від морських об'єктів.

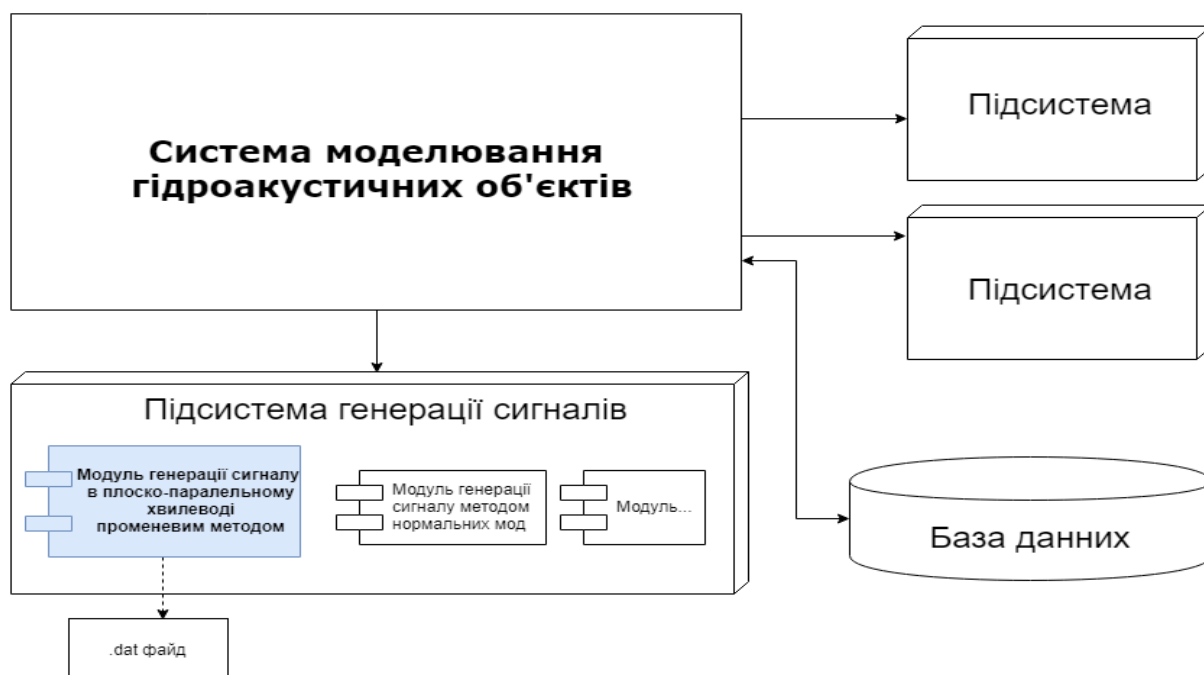


Рисунок 1 — Можливе місце модуля в системі

Оскільки розроблений програмний продукт може використовуватися підсистемою комплексу моделювання гідроакустичних процесів модуль повинен мати:

- один вхід і один вихід. На вході програмний продукт отримує визначений набір початкових даних, виконує їх обробку і повертає один набір вихідних даних;

- функціональну завершеність. Модуль виконує набір визначених операцій для реалізації визначених операцій для реалізації кожної окремої функції, достатніх для завершення початої обробки даних;

- логічну незалежність. Результат роботи даного фрагменту програми не залежить від роботи інших модулів;

- слабкі інформаційні зв'язки з іншими програмними модулями. Обмін інформацією між окремими модулями повинен бути мінімальним.

## 2 ОПИС ПРЕДМЕТНОЇ ОБЛАСТІ

У даному розділі наведені теоретичні відомості про гідроакустику, сфери застосування, проблематика, види задач та інше, що входить до даної області, було розглянуто існуюче рішення для .

### 2.1 Гідроакустика

Гідроакустика — розділ акустики, що вивчає випромінювання, прийом і поширення звукових хвиль в реальному водному середовищі (в океанах, морях, озерах і т. д.) для цілей підводної локації, зв'язку і т. ін.

Головна особливість підводних звуків — їх мале загасання, внаслідок чого під водою звуки можуть поширюватися на значно більші відстані, ніж, наприклад, у повітрі [1].

Крім загасання, обумовленого властивостями самої води, на дальність поширення звуків під водою впливають рефракція звуку, його розсіяння і поглинання різними неоднорідностями середовища.

Інтенсивні дослідження процесів поширення акустичних хвиль у двовимірних і тривимірних підводних неоднорідних хвилеводах отримали значний імпульс, починаючи з 80-х років минулого століття в зв'язку з потребами дистанційного зондування і акустичного моніторингу регіонів Світового океану. Практично у всіх видах сигналізації, зв'язку, локації, дистанційного дослідження водних мас і дна океану використовуються звукові хвилі [2].

При проектуванні інформаційно-вимірювальних систем також застосовуються результати моделювання хвильових процесів та надають змогу вирішити широке певні взаємопов'язані задачі, такі як формування певних структур акустичних полів в заданих хвилеводах, створенні гідроакустичних антен та ін. Тому проблеми створення і вдосконалення математичних моделей акустичних процесів в океані є

досить актуальними, також є необхідність в розробці спеціалізованих програмних продуктів.

## 2.2 Технології гідроакустики

Звукові хвилі — єдиний вид хвиль, які мають можливість розповсюджуватися в морському середовищі без великого ослаблення на значні відстані, тому на їх основі створюються дистанційні методи і технології вивчення рельєфу дна, дослідження складу і різноманітних властивостей морських ґрунтів і порід, визначення глибин і виявлення рухомих і нерухомих об'єктів, пошук місцезнаходження затонулих об'єктів за заданими характеристик, перевірка стану підводних гідротехнічних споруд, дослідження стану підводних частин гідротехнічних споруд, дослідження середовища та ін. [3].

В більшості гідроакустичні системи мають дуже схожі складові та використовуються у вирішенні багатьох задач, що призводить до зацікавленості в взаємодії між різними організаціями, які так або інакше, працюють в напрямках застосування та дослідження гідроакустичних технологій. [4].

Гідроакустичні засоби (датчиків) створюються для виконання певних цілей, наприклад:

- геоакустичні;
- знаходження ресурсів(сировинних і біологічних);
- моніторингу стратифікації морського середовища;
- моніторинг експлуатації та стану підводних трубопроводів вуглеводневої сировини;
- боротьба з морськими мінами;
- розпізнання підводних ворожих сил;
- робота гідроакустичних станцій та систем підводних та надводних кораблів;
- робота гідроакустичних систем розвідки ситуації в різних підводних зонах.

Методи гідроакустики приміняються та реалізуються за допомогою пристроїв та систем. Гідроакустичні пристрої це сукупність елементів, принципи роботи яких базуються на використанні властивостей акустичних хвиль у водному середовищі[5].

Враховуючи особливості водного середовища в моніторингу використовують досить цікаві методи. Гідроакустика передбачає запис сигналів, які показують зміни акустичного тиску води, що створюються звуковими хвилями. У воді утворюється шар де звуковий рух є більш повільним, але дуже ефективним, даний шар є звукоізоляційним підводним каналом, який зазвичай знаходиться на глибині близько 1000 м, та утворює своєрідний хвилевід. Вище та глибше від хвилеводу швидкість звуку змінюється через вплив температури, а в середині — сигнали відбиваються від стінок каналу назад в його межі і таким чином поширюються на дуже великі відстані. Також гідроакустика є однією з чотирьох технологій, що використовуються міжнародною системою моніторингу для перевірки дотримання договору про всеосяжну заборону ядерних випробувань. Ядерні вибухи під водою, в атмосфері, біля поверхні океану або під землею біля берегової лінії генерують звукові хвилі, які можуть бути виявлені на відстані в сотні тисяч кілометрів. Як наслідок, всього одинадцять станцій є достатніми для моніторингу великих океанів Землі, з акцентом на південній півкулі, де в основному домінує вода.

Для прикладу розглянемо принцип роботи гідролокатора(рисунок 2.1), що заснований на вимірюванні часу, протягом якого звуковий імпульс проходить від випромінювача до об'єкта, а його ехосигнал, відбивається з певним імпульсом, коли зустрічає на своєму шляху об'єкт дослідження. При відомому часі проходження імпульсу від випромінювача до об'єкта, часі проходження ехосигнала від об'єкта до приймача та швидкості поширення звуку у воді можна визначити відстань між ними. Метод визначення відстані між об'єктами в воді по часу проходження звукового імпульсу застосовується в різноманітних акустичних приладах.

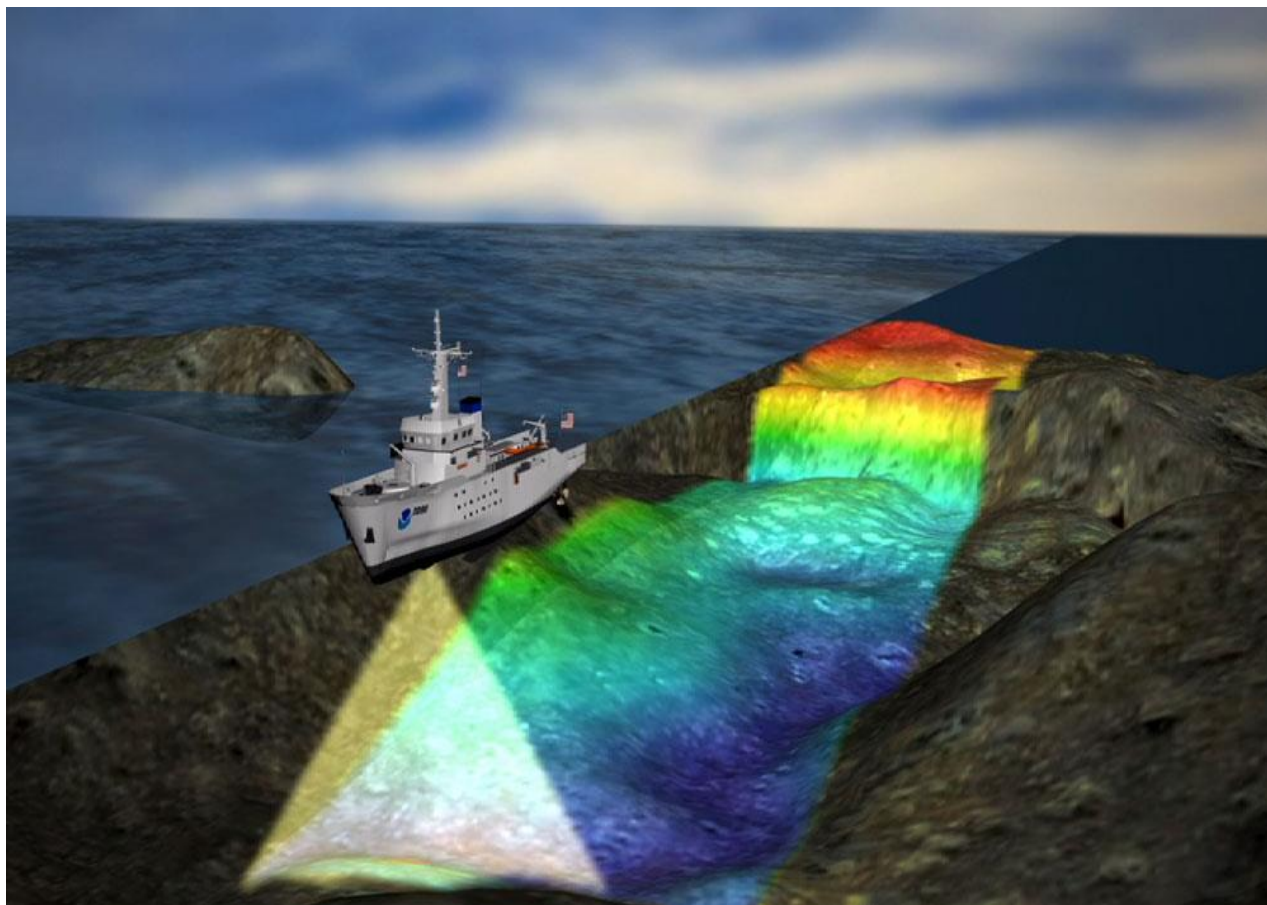


Рисунок 2.1 — Принцип роботи гідролокатора

Залежно від призначення гідролокатори мають різноманітні конструктивні виконання: для дистанційного зондування складу верхнього шару ґрунту, для сейсмічного зондування, для огляду дна, навігаційні системи для визначення координат, для пошуку і визначення координат рибних скупчень, що встановлюються на видобувних судах та багато іншого.

Очевидно, що гідроакустика знайшла широка застосування, та має високий пріоритет в наукових дослідженнях.

### **2.3 Програмні продукти для моделювання сигналів Actran та Adams**



На даний момент, універсальним програмним продуктом який використовується, в переважній більшості випадків для моделювання сигналів є Actran(рисунок 2.2).

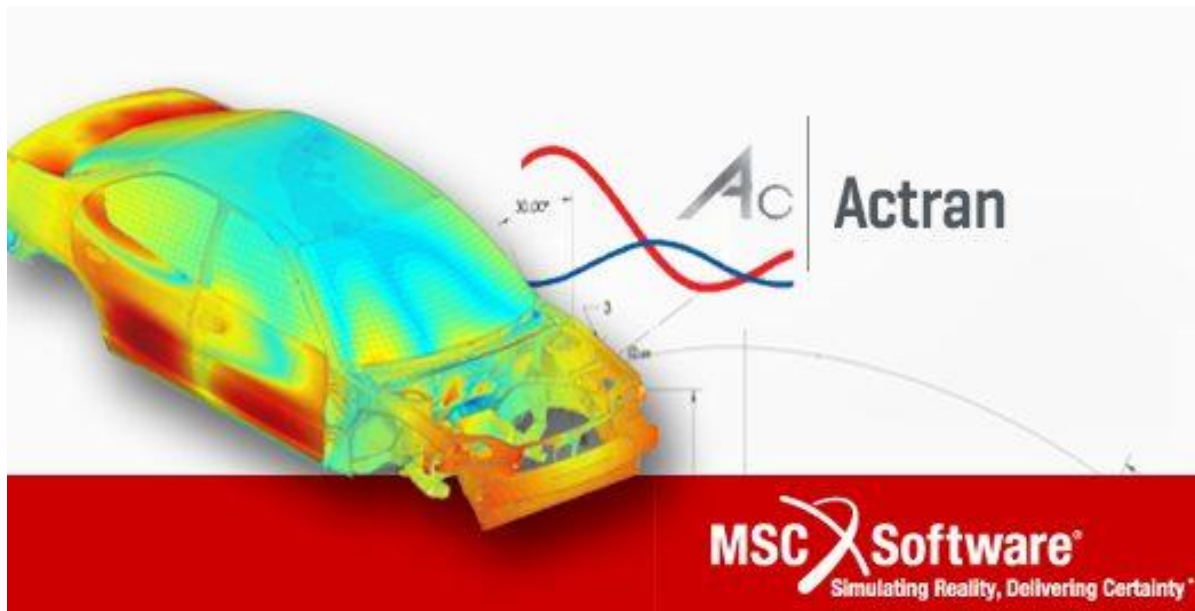


Рисунок 2.2 — Логотип Actran

Розробка Actran почалася, в наш час, професором Луврської інженерної школи і Жан-Луї Міхеот та професор в університеті Брюсселя і колишній президент Королівської академії наук Жан-П'єром Котом. Початкова ідея полягала в розробці інструментального засобу моделювання джерел звуку на основі скінченних елементів для віброакустичних застосувань, які могли б подолати обмеження, в той час, домінуючого методу скінченних елементів. Використання нескінченних елементів дозволило моделювати складні джерела шуму, поєднання декількох матеріалів в одній моделі і обробку моделей з кількома мільйонами ступенів свободи. Початковим цільовим додатком було прогнозування акустичної передачі через складні перегородки. Центральною особливістю Actran було використання нескінченних елементів, як альтернативу скінченних елементів для моделювання невідповідних граничних умов і обчислення далекого поля[13].

Ранній розвиток фінансувався промисловим консорціумом, а перший комерційний випуск був широко доступний у 2002 році.

Astran розроблена на мовах python і C ++ і є сумісним з операційними системами Linux та Windows.

В основі Astran лежать методи скінченних і нескінченних елементів, що дозволяють імітувати акустичні ефекти як в закритих обсягах (порожнинах), так і в відкритому просторі, враховуючи умови не відбиття звуку на межах області, що розраховується. Багата бібліотека елементів і матеріалів надає широкі можливості моделювання. Основу бібліотеки елементів складають одновимірні, двовимірні та тривимірні елементи для розв'язування тривимірних, двовимірних і осесиметричних задач, багатоточкових з'єднань, нескінченних елементів і т.д. Бібліотека представлених матеріалів надає можливість моделювання акустичних середовищ (газів, рідин) з урахуванням затухання і в'язко-термічних втрат в тонких шарах[13].

Як зовнішні навантаження та граничні умови, розглядаються джерела шуму різних типів, моди коливань акустичного середовища в каналах постійного перерізу, граничні умови по тиску, швидкість, прискорення, імпедансні граничні умови, потужність і кінематичні впливи на структуру в склад розрахункової моделі, структурні граничні умови, випадкові флуктуації тиску акустичної середовища, об'ємні та поверхневі джерела на основі акустичних аналогій.

Аналіз акустики виконується в основному в частотній області методом прямого інтегрування. Поряд з цим підтримується розрахунок випадкових ефектів, розрахунок власних частот і форм, розрахунок частотної характеристики модальним методом або комбінація модальних і прямих методів для акустичного середовища і структури. Також підтримується розрахунок перехідних акустичних процесів у часовій області.

Програма Astran в даний час розділена і ліцензована в різні модулі в залежності від цільової програми та фізики, що беруть участь:

— модуль Acoustics: основний модуль для аналізу акустичного випромінювання. Типовими застосуваннями є: шумове випромінювання від силових агрегатів, передача шумів через глушники та глушники;

— модуль VibroAcoustics: модуль, присвячений сильно зчепленому віброакустичному моделюванню Типовими програмами є: звукова передача через структури (стіни, вікна тощо), динаміки, підводна акустика;

— модуль AeroAcoustics: модуль, присвячений обчислювальній аероакустиці. Типовими застосуваннями є HVAC-канали, відцентрові та осьові вентилятори, шум від бокового вікна;

— модуль TM: модуль, присвячений турбомашинному шуму. Типовими застосуваннями є турбореактивні двигуни;

— DGM: модуль рішення лінійних рівнянь Ейлера. Цей модуль явно розв'язує часовий домен, а числова схема — це розривний метод Галеркіна.

— користувацький інтерфейс, загальний для всіх модулів. Він використовується для попереднього обробки модулів Actran, включаючи генерацію та модифікацію акустичних сіток і пост-обробку результатів.

Важливою особливістю Actran є його близька інтеграція з MSC Nastran, яка дозволяє ефективно використовувати позитивні сторони кожного з продуктів. Аналіз коливань конструкцій проводиться за допомогою MSC Nastran, а моделювання розповсюдження звуку за допомогою Actran. За допомогою додаткових модулів реалізована інтеграція з системою Adams[13].

Adams — це комплексне рішення для створення віртуальних прототипів складних динамічних систем(рисунок 2.3) та включає в себе:

— широкий набір видів кінематичних зв'язків, пружних і дисипативних ланок з лінійними і нелінійними характеристиками, навантажень, кінематичних впливів і т.д., доступних користувачеві для побудови розрахункової моделі в максимальному ступені що відтворює властивості реального виробу;



Рисунок 2.2 — Adams

— легкість вивчення і використання, так як дослідження віртуального прототипу в Adams відповідає основним етапам роботи з досвідченим зразком виробу (розробка - випробування - вдосконалення);

— дружній, інтуїтивно зрозумілий інтерфейс - якщо інженер знайомий з іншими програмними засобами CAE або CAD, то швидко освоїть роботу і з Adams;

— ефективні засоби візуалізації результатів моделювання, включаючи анімацію і побудова графіків;

— можливість параметризації розрахункової моделі — модифікація параметрів призводить до автоматичної зміни властивостей моделі та / або її зміни, параметри моделі можуть бути пов'язані функціональними залежностями і т.п. ;

Завдяки роботі з великим числом як вхідних так і вихідних форматів даних, результати моделювання і розрахунку в Adams можна з успіхом використовувати в КЕ системах;

— наявність «вертикальних додатків» (надбудов для Adams), таких як Adams / Car, Adams / TrackedVehicle, Adams / Machinery, дозволяє розширити спектр вирішуваних завдань, спростити взаємодію інженера з розрахунковою моделлю, як на етапі її створення і модифікації, так і на етапі проведення розрахункових досліджень;

— застосовуючи комплексний підхід при моделюванні виробів в Adams, з використанням пружних деталей і компонентів, результати розрахунків динаміки можна використовувати в якості історій навантаження і навантажувальних циклів (DutyCycle) для подальшого дослідження довговічності в системі MSC.Fatigue;

— наявність додаткового інтерфейсу для взаємодії з зовнішніми розрахунковими пакетами і проведення спільних розрахунків відкриває можливості до моделювання складних механічних моделей з високо нелінійними (як геометрично так і фізично) пружними ланками, підготовленими в програмному комплексі Marc;

— сумісність з програмними засобами для моделювання систем автоматичного регулювання та керування Easy5, Simulink, Functional Mock-up Interface (FMU) і з одними програмами забезпечує моделювання і дослідження складних гетерогенних динамічних систем і пристроїв;

— додаткові розширення - плагіни надають інженеру можливості по прямої взаємодії (з інтерфейсу Adams) з KE системою MSCNastranEmbeddedFatigue для оцінки довговічності вузлів модельованого виробу (розширення Adams2NEF), з системою Actran для оцінки акустичних властивостей (плагін Adams2Actran).

### 3 МОДЕЛЮВАННЯ ГІДРОАКУСТИЧНОГО СИГНАЛУ

Для моделювання сигналу потрібно добре розуміти алгоритм розрахунку його параметрів, тому огляд математичної частини є важливим моментом. Для вирішення задачі було використано променевий метод при плоско-паралельних границях хвилеводу.

#### 3.1 Математичні моделі хвильових процесів

З математичної точки зору поширення гармонійних акустичних хвиль описується крайовими задачами для рівняння Гельмгольца з комплексним несамопряженим оператором.

В даний час для комп'ютерного моделювання акустичних полів в підводних неоднорідних хвилеводах використовуються три класи математичних моделей у вигляді крайових (початково-крайових) завдань для еліптичних (параболічних типу Шредінгера) хвильових рівнянь [6];

— крайові задачі для еліптичного хвильового рівняння Гельмгольца з комплекснозначним несамопряженим оператором;

— задача Коші для еліптичного перетвореного хвильового рівняння Гельмгольца;

— початково-крайові задачі для параболічних хвильових рівнянь типу Шредінгера з комплекснозначним несамопряженим оператором.

Для неоднорідного середовища акустичне поле точкового гармонійного джерела (Залежність від часу приймається у вигляді  $e^{-i\omega t}$ ) в прямокутній системі координат  $(x, y, z)$  описується рівнянням Гельмгольца:

$$p(x) \operatorname{div} \left( \frac{1}{p(x)} \operatorname{grad} p \right) + k_0^2 (n^2(x) + i v(x)) p = -\delta(x - x_0), \quad (3.1)$$

де  $p(x), x = (x, y, z)$  — акустичний тиск;  $k_0 = \omega/c_0$  — хвильове число;  $n(x) = \frac{c_0}{c(x)}$ ,  $x = x(x, y, z)$  — коефіцієнт заломлення;  $c(x)$  — швидкість звуку ( $c_0$  — швидкість звуку в деякій точці);  $p(x)$  — щільність середовища;  $v(x) \geq 0$  — коефіцієнт затухання;  $x_0 = (x_0, y_0, z_0)$  — координати джерела;  $\delta(x - x_0)$  — дельта функція Дірака;  $i = \sqrt{-1}$  — уявна одиниця [8].

В циліндричній системі координат  $(r, \varphi, z)$  рівняння (3.1) приймає вигляд:

$$\begin{aligned} \frac{p}{r} \frac{\partial}{\partial r} \left( \frac{r}{p} \frac{\partial p}{\partial r} \right) + \frac{p}{r^2} \frac{\partial}{\partial \varphi} \left( \frac{1}{p} \frac{\partial p}{\partial \varphi} \right) + p \frac{\partial}{\partial z} \left( \frac{1}{p} \frac{\partial p}{\partial z} \right) + k_0^2 n^2(r, \varphi, z) + i v(r, \varphi, z) p = \\ = - \frac{\delta(r)}{2\pi r} \delta(z - z_0). \end{aligned} \quad (3.2)$$

При моделюванні хвильових задач широко використовують математичні моделі м'яких, жорсткою і імпедансові кордонів хвилеводів. Імпедансі умови містять комплексні коефіцієнти і описують втрати акустичної енергії. У неоднорідному за щільністю середовищі на кордоні розділу середовищ виконуються умови безперервності акустичного тиску і потоку. Крім того, для коректної постановки крайових задач в необмежених областях на нескінченності необхідно задати умови випромінювання, що забезпечують єдність розв'язку. В загальному випадку простору  $R^n$  для вирішення рівняння Гельмгольца в середовищах без згасання ( $v(x) = 0, n(x) = 1$ ) повинні виконуватися умови випромінювання Зоммерфельда:

$$p(x) = O\left(r^{\frac{1-n}{2}}\right), \frac{\partial p}{\partial r} \mp i k_0 p = o\left(r^{\frac{1-n}{2}}\right), x = (x, y, z), r = |x| \rightarrow \infty, \quad (3.3)$$

де знаки  $\mp$  відповідають залежності з часом  $e^{\mp i \omega t}$  [8].

### 3.2 Перехід до двовимірних координат

Для моделювання сигналу променевим методом, необхідно перейти до двовимірної системи координат. Даний перехід буде залежати від координат джерела звуку та ГАС, що його приймає. Припустимо, що джерело сигналу завжди буде лежати в координатах  $(0, 0, z_0)$ , а система, що приймає сигнал в координатах





При цьому нова система координат буде мати вісі  $(r, z)$ , де значення  $z$  залишиться таким самим як і в тривимірних координатах, а  $r(x, y)$  буде залежати від координат об'єкту по  $x$  та  $y$ . Модельна система координат лежить в площині між точками положення джерела, гідроакустичної система приймача та буде паралельною до вісі  $z$ . Джерело отримає координати:

$$r = \sqrt{0^2 + 0^2} = 0$$

$$z_0 = z_0$$

Координати ГАС:

$$\begin{cases} r = \sqrt{x^2 + y^2} \\ z = z \end{cases}$$

### 3.3 Однорідний океан постійної глибини

Найпростіший профіль швидкості звуку має вигляд  $c(z) = c_0$ , де  $c_0$  — константа. В цьому випадку для точкового джерела граничне рівняння прийде до вигляду

$$\Delta p + k^2 p = -\frac{\delta(r)}{2\pi r} \delta(z - z_0). \quad (3.4)$$

В даному випадку координати джерела  $x_0$  це  $r = 0, z = z_0$ . Граничною умовою на поверхні є рівність:

$$p = 0 \text{ при } z = 0. \quad (3.5)$$

Якщо глибина постійна ( $h = \text{const}$ ), граничною умовою дна буде:

$$p_z = 0 \text{ при } z = -h. \quad (3.6)$$

Крайові задачі (3.4) - (3.6) з правильною умовою випромінювання визначають поле тиску монохроматичного точкового джерела в однорідному океані постійної глибини з вільною поверхністю та твердим дном. Оскільки задача циліндрично-симетрична, отримане рішення  $p(r, z)$  не буде залежати від кутової координати  $\theta$ [9].

### 3.4 Променеве представлення

Досить наочним вираженням для  $p(r, z)$  є променеве представлення. Для цього розглянемо спочатку рівняння (3.4) в трьохвимірному просторі, ігноруючи граничні умови (3.5) і (3.6). Загальне сферично-симетричне рішення рівняння (3.4) має вигляд  $p_0(R) = Ae^{ikR}/R_0 + Be^{-ikR}/R_0$ , де  $A + B = 1/4\pi$ , а  $R_0 = [r^2 + (z - z_0)^2]^{1/2}$  — відстань від джерела. Щоб вилучити хвилю  $e^{-ikR}$ , що приходить з безкінечності, накладемо умову випромінювання:

$$\lim_{R \rightarrow \infty} R_0[p'_0(R) - ikp_0(R_0)] = 0 \quad (3.7)$$

З цієї умови отримуємо, що  $B = 0$  і сферично симетричне рішення (3.8) прийме вигляд

$$p_0(R_0) = e^{ikR}/4\pi R_0 \quad (3.9)$$

Показник експоненти в (3.9) можна інтерпретувати як перемноження  $ik$  і фазової функції  $R$ . Ця фаза рівна нулю в джерелі і зростає пропорційно відстані вздовж прямих ліній, які з'єднують джерело з точкою поля. Ми назвемо цю пряму лінію променем. Множник  $1/R_0$  називається амплітудою. Він спадає протилежно пропорційно квадратному кореню з площі поперечного перерізу променевої трубки, оскільки ця площа зростає як  $R_0^2$ . Коли певні промені падають на верхню границю  $z = 0$ , тоді виникають відбиті промені. Амплітуду відбитої хвилі потрібно помножити на коефіцієнт відбиття, рівний  $-1$ , для того щоб сума падаючої і відбитої хвиль відповідала умові (3.5). Так як всі промені виглядають, наче прийшли від уявного джерела, яке знаходиться в точці  $r = 0$ ,  $z = -z_0$ , то задана побудова дасть відбиту хвилю вигляду  $-e^{ikR'}/R'$ , де  $R'$  це відстань від уявного джерела (рисунок 3.2).

Аналогічна побудова відбувається для променів відбитих від дна. Але тут коефіцієнт відбиття рівний  $+1$ , оскільки сума падаючої і відбитої хвиль мають задовольняти умову (3.6). Уявне джерело буде розміщене в точці  $z = -z_0 - 2h$ . Тому відбита від дна хвиля матиме вигляд  $e^{ikR''}/R''$ , де  $R''$  це відстань від уявного джерела.

Багаторазове відбиття об стінки хвелеводу породить групи променів, кожна з яких виглядає, наче прийшла з деякої уявної точки  $z = \mp z_0 + 2nh$ ,  $n = 0, \mp 1, \dots$ . Враховуючи кількість відбиттів, приходимо до виразу загального поля  $p$  в вигляді суми падаючої хвилі та відбитих хвиль:

$$p(r, z) = \frac{1}{4\pi} = \sum_{n=-\infty}^{\infty} (-1)^n \left\{ \frac{e^{ik[r^2 + (z-z_0-2nh)^2]^{\frac{1}{2}}}}{[r^2 + (z-z_0-2nh)^2]^{\frac{1}{2}}} - \frac{e^{ik[r^2 + (z+z_0-2nh)^2]^{\frac{1}{2}}}}{[r^2 + (z+z_0-2nh)^2]^{\frac{1}{2}}} \right\}. \quad (3.10)$$

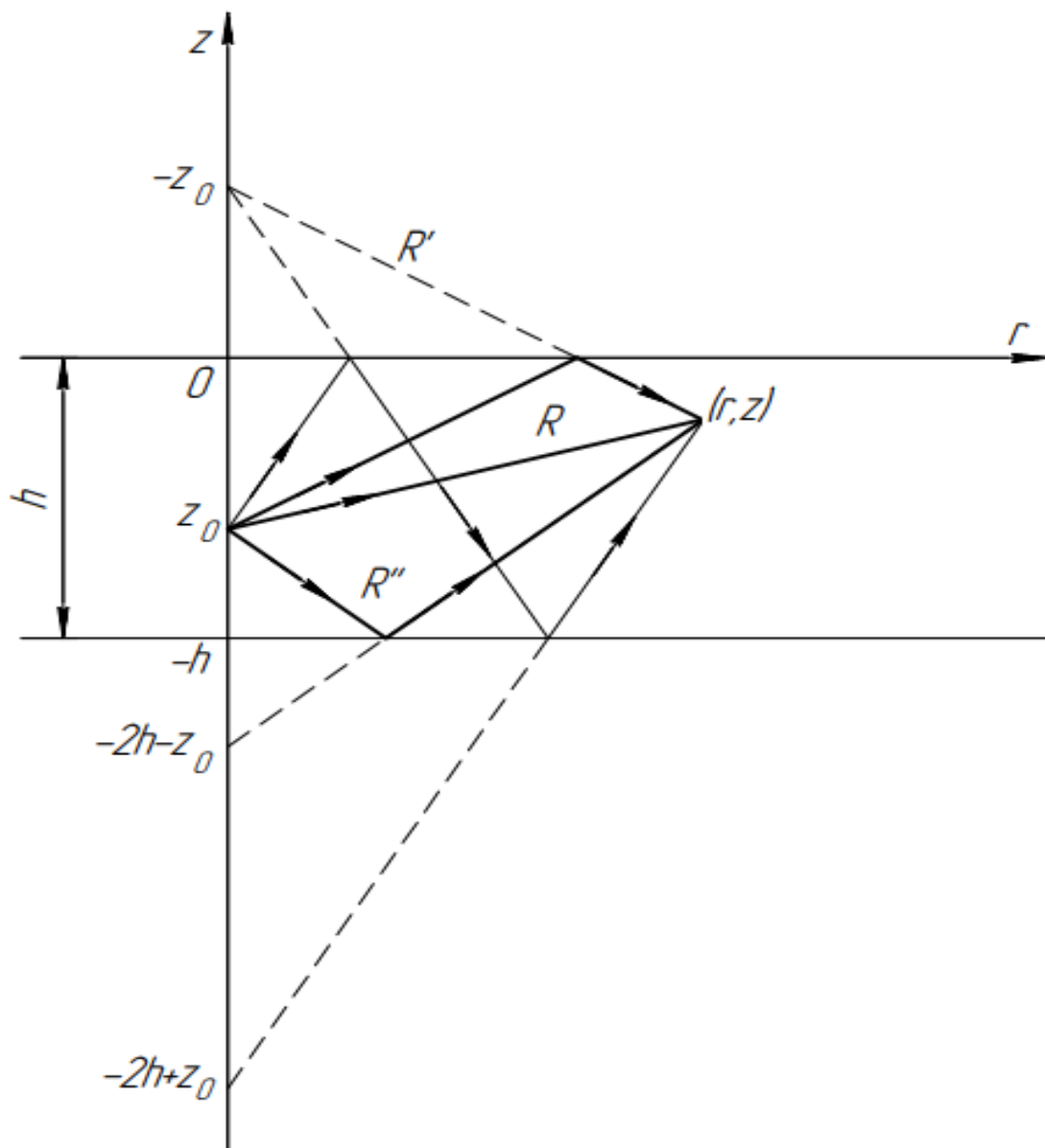


Рисунок 3.2 — Координати уявних джерел

Це і є променеве представлення рішення  $p(3.10)$ , яке відповідає багаторазовому відбиттю, тому що в ньому кожен доданок, окрім члена з  $n = 0$ , описує хвилю, що відбилась деяку кількість раз від поверхні та дна хвилеводу[9].

На наведеному Рисунку 3.2 зображено точкове джерело, заходиться в точці  $z = z_0$ , випускає промені у всіх напрямках. Чотири промені попадають в точку  $(r, z)$ . Перший промінь довжини  $R$  є прямим.

Другий промінь довжини  $R'$  відбився від границі поверхні  $z = 0$  та має вигляд, наче він прийшов з джерела, що знаходиться в точці  $z = -z_0$ . Третій промінь довжини  $R''$  відбився від дна і має вигляд, наче прийшов з джерела в точці  $z = -2h - z_0$ . Четвертий промінь відбився спочатку від поверхні хвилеводу, а потім від дна і виглядає як промінь, що прийшов з джерела в точці  $z = -2h + z_0$ . Таким чином розраховується довжина  $n$ -го променю.

### 3.5 Розрахунок коливальної швидкості

Для моделювання сигналу, окрім сумарного акустичного тиску, розраховується ще один фізичний параметр звукового поля — коливальна швидкість.

При розповсюдженні звукової хвилі, частинки повітря починають коливатися відносно положення рівноваги. Швидкість з якою коливаються частинки середовища відносно свого положення рівноваги, називається коливальною. Коливальна швидкість є похідною величиною від акустичного тиску, якщо акустичний тиск обчислюється за допомогою функції косинуса, то коливальна швидкість є функцією синуса та обчислюється за формулою:

$$v_{x,y,z} = \sin(\omega t + \varphi(R, \lambda) \frac{|v_{x,y,z}|}{R_0}) \quad (3.11)$$

де  $\omega$  — кутова частота,  $t$  — час,  $\frac{|v_{x,y,z}|}{R_0}$  — відношення довжини проекції коливальної швидкості на відповідну вісь до відстані від джерела до ГАС або

довжини хвилі що пройшла напряму від джерела до гідроакустичної системи без відбиттів,  $\varphi(R, \lambda)$  моментальне значення сумарної фази для поля сигналу.

Для знаходження модулю коливальної швидкості, розраховується значення фази для кожного променя, та її сумарне значення (рисунк 3.3). Гармонічні коливання можуть бути отримані з обертання вектору коливальної швидкості по колу. Кут повороту коливального вектора для  $n$ -го променя в момент попадання до приймача і буде його фазою. Наприклад, протягом одного періоду фаза змінюється від 0 до  $2\pi$ . Це відповідає тому що вектор зробить повний оберт проти часової стрілки, а для  $n$ -го променя “надлишкова” фаза розрахується, як залишок від ділення довжини даного променя на довжину хвилі.

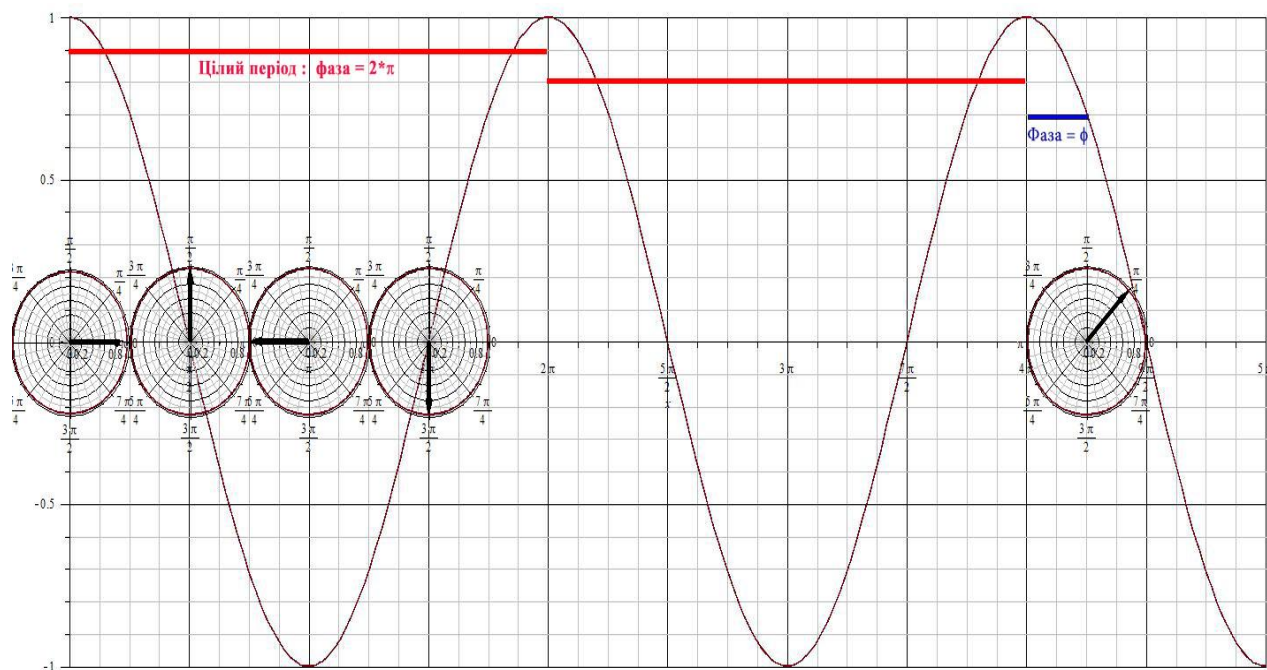


Рисунок 3.3 — Пояснення до обрахунку фази

Розрахунок модулю коливальної швидкості буде векторним та розрахується за правилом паралелограма, де довжина кожного вектору буде нормуватися за амплітудою, а кут прийме значення фази (рисунк 3.4).

Після обрахунку коливальних швидкостей та сумарного тиску, отримані дані будуть характеризувати гідроакустичне поле сигналу в межах акваторію за заданими параметрами.

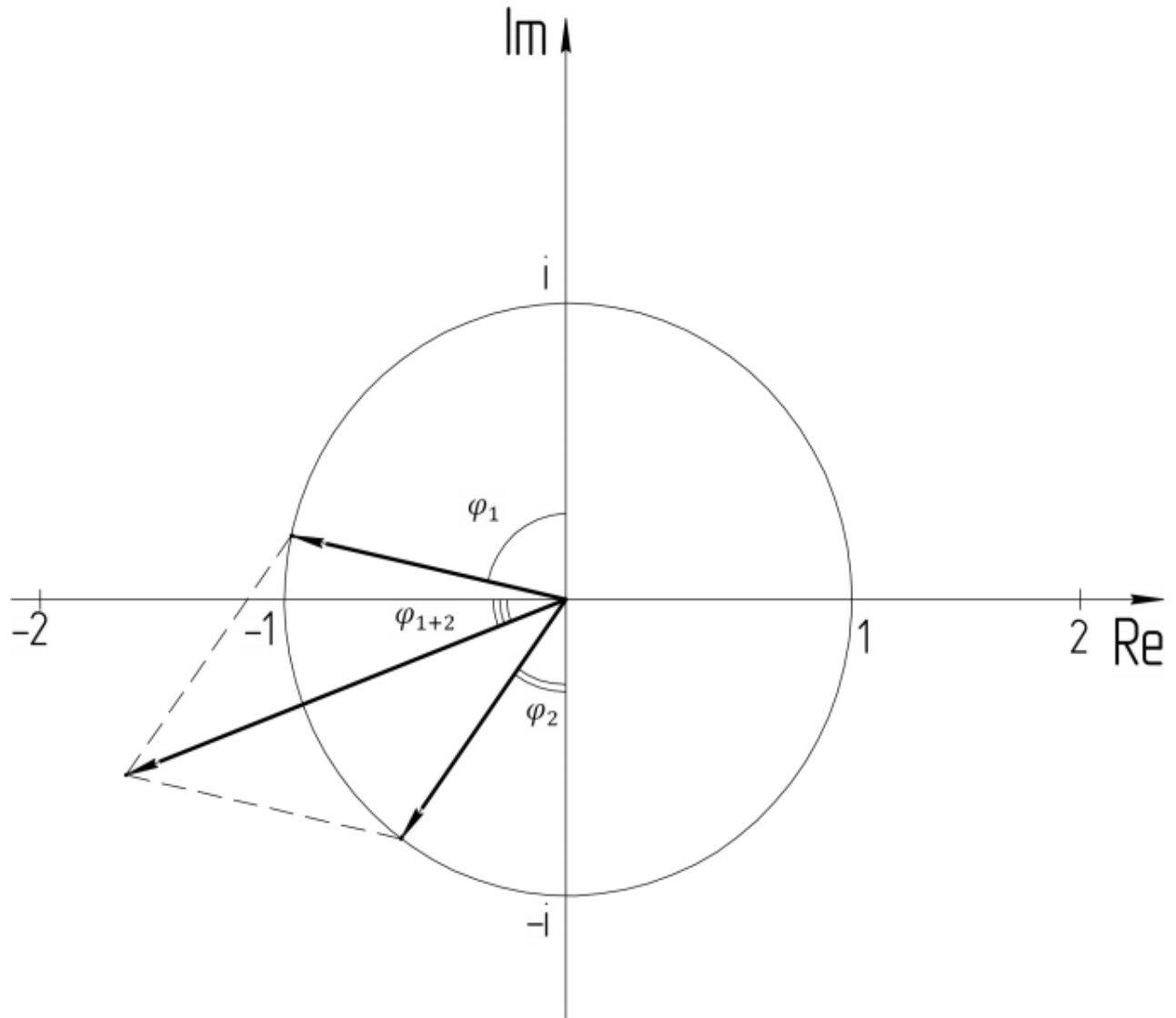


Рисунок 3.4 — Розрахунок модулю коливальної швидкості

Даний метод ефективний в ближній зоні сигналу. Ближня зона — область поля поблизу джерела сигналу, в якій спостерігаються зумовлені інтерференцією немонотонні зміни амплітуди поля. Більше 80% випромінюваної джерелом енергії знаходиться в її межах.

### 3.6 Альтернативний метод моделювання сигналу

Для початку, в моделюванні поширення звукових хвиль слід визначити хвилевод, розділений двома горизонтальними положеннями і двома вертикальними позиціями. Часто вертикальні позиції вважаються морською поверхнею і морським дном. Положення джерела акустичного сигналу встановлюється в цьому діапазоні.

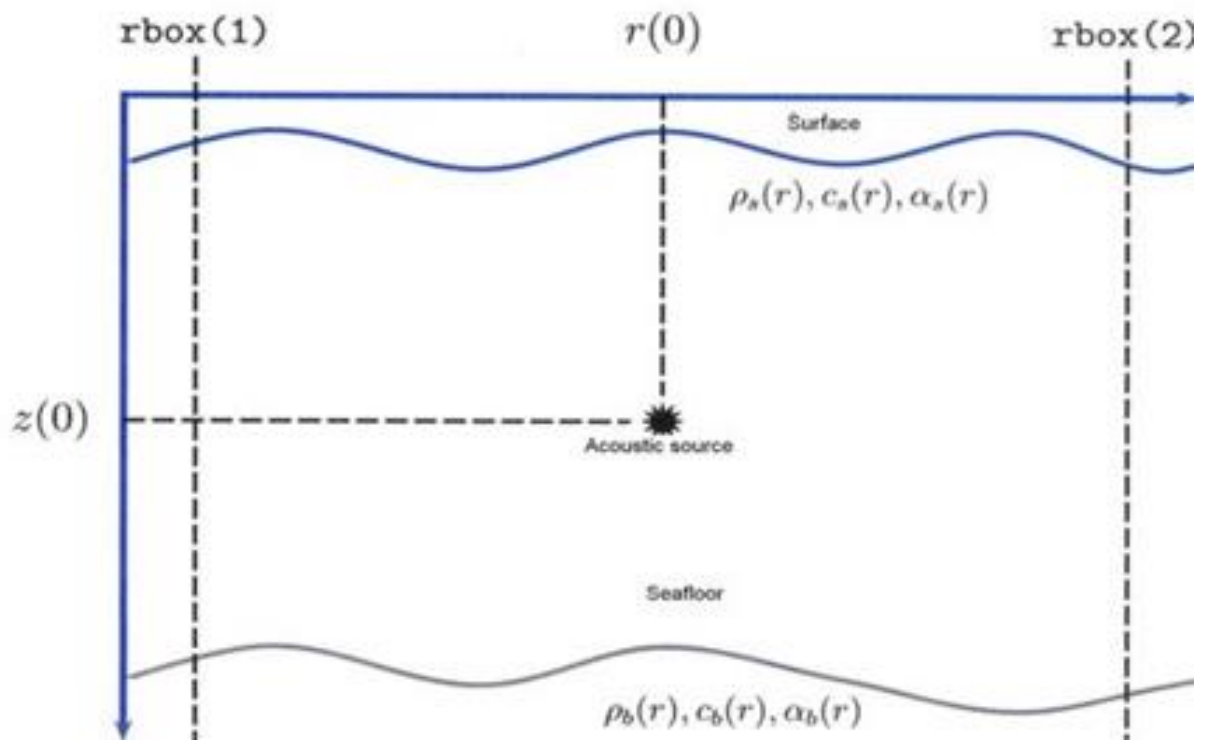


Рисунок 3.5 — Трасування променів водного діапазону

З точки зору променевої теорії підійшла задача моделювання поширення звукових хвиль. Загальна стратегія трасування променів полягає в наступному:

- встановити кути запуску в положенні джерела, а потім простежити промінь для кожного кута запуску, доки він не поглинеться або вийде з діапазону;
- на кожному кроці визначають координати променів, перевіряють, чи є промінь над поверхнею і нижче дна. Для позитивної перевірки визначають перетин

променя з де-межею, відображають промінь і продовжують з визначенням координат променів.

Для застосування цієї стратегії слід знати профіль швидкості звуку в розглянутому діапазоні і потім визначити координати променів.

Профіль швидкості звуку можна визначити шляхом прямого вимірювання або з використанням методу розрахунку на основі датчика CTD - провідності (солоності), температури, глибини. Аналізуючи результати ряду розрахункових методів (рівняння ЮНЕСКО-Чен і Міллера, рівняння Медвіна, рівняння Коппенса), рівняння Леруа наближається до кращих значень швидкості звуку.

Щоб простежити гідроакустичні промені в певному діапазоні згідно з теорією променів, необхідно розв'язати систему рівнянь Ейконала, яка регулює поширення акустичних хвиль.

Проблеми розповсюдження акустичних променів часто включають пунктуальні джерела, які мають циліндричну симетрію, або, принаймні, вважаються середовищами, які повільно змінюються по азимуту і азимутній залежності можна ігнорувати при обчисленні передачі в певному напрямку[15]. У цьому випадку  $r = (r, z)$  де  $r$  — горизонтальна координата, а  $z$  — вертикальна координата (глибина) ) [12].

У циліндричних координатах рівняння, які регулюють поширення акустичних променів, приймуть вигляд:

$$\begin{aligned}\frac{dr}{ds} &= c\sigma_r(s), \frac{d\sigma_r}{ds} = \frac{1}{c^2} \frac{dc}{dr} \\ \frac{dz}{ds} &= c\sigma_z(s), \frac{d\sigma_z}{ds} = \frac{1}{c^2} \frac{dc}{dz}\end{aligned}$$

де  $r(s)$  і  $z(s)$  представляють променеві координати в циліндричних координатах, а  $s$  — довжина дуги вздовж променя. Пара  $(\sigma_r, \sigma_z)$  являє собою компоненту повільності  $\sigma = \frac{1}{c^2}$ , де  $c$  — швидкість звуку.

Початкові значення координат  $r(s)$ ,  $z(s)$ ,  $\sigma_r(s)$  та  $\sigma_z(s)$  вздовж кривої  $[r(s), z(s)]$  будуть:



$$\tau = \int_r \frac{ds}{c(s)}$$

Коли швидкість звуку на глибині залежить тільки від вертикальної компоненти повільності:

$$\sigma_z(s) = \frac{\cos\theta(s)}{c(s)}$$

це буде константою у вигляді відношення:

$$\frac{d\sigma_z}{ds} = 0$$

Для визначення чисельного розв'язання системи рівнянь Ейконала ми починаємо представивши систему у вигляді векторного рівняння[14]:

$$\frac{dy}{ds} = f$$

Де  $f$  та  $y$ :

$$y = \begin{bmatrix} r \\ z \\ \sigma_r \\ \sigma_z \end{bmatrix}, f = \begin{bmatrix} c\sigma_r \\ c\sigma_z \\ -\frac{1}{c^2} \frac{dc}{dr} \\ -\frac{1}{c^2} \frac{dc}{dz} \end{bmatrix}$$

Векторне рівняння вирішується за допомогою чисельного методу Рунге - Кутта - Фельберга.[12]

Для відстеження гідроакустичних променів у певному морському діапазоні необхідно встановити ряд параметрів. Параметри структуровані в три блоки[16]:

— параметри, що характеризують морський діапазон:

- горизонтальний діапазон;
- вертикальний діапазон;
- тип поверхні моря: корпуси;

— абсорбуюча плоска поверхня;

— відмінно відображає плоску поверхню;

- типи судових поверхонь;

— абсорбуюча плоска поверхня;

— відмінно відображає плоску поверхню;

— параметри, що характеризують джерело акустичного сигналу (ми розглядали випадок точного джерела):

- вихідна горизонтальна координата;
- вертикальна координата джерела;
- частота джерела;
- кількість промальованих променів;
- перший кут запуску;
- останній кут запуску;
- параметри, що характеризують профіль швидкості звуку.

Для прикладу моделювання сигналів даним методом наведені результати для умов Чорного моря. Чорноморський басейн має деякі специфічні характеристики, такі як взаємодія верхніх і нижніх шарів, що обумовлює нерівномірну структуру морського середовища. Температура і солоність водних шарів в зоні континентальної платформи, що відповідає румунському берегу, не залишаються постійними протягом усього року. Варто відзначити, що середня низька мінералізація вказує на весь басейн Чорного моря, але з граничними значеннями в околицях гирла річок (явище розведення).

Існування значних відмінностей між профілями швидкості звуку взимку в порівнянні з тим, що влітку, в тому ж географічному положенні, визначається як, для аналізу поширення гідроакустичних хвиль для двох специфічних профілів швидкості звуку літо / зима в одному області. Два профілі швидкості звуку були розраховані за допомогою рівняння Леруа. Значення температури, солоності та глибини вимірювали за допомогою датчика CTD (рисунок 3.6).

На наступних малюнках представлені результати моделювання поширення гідроакустичного сигналу з літнім та зимнім профілями швидкості звуку за допомогою програмних модулів, розробленими для трасування променів (рисунки 3.7 — 3.8) [12].

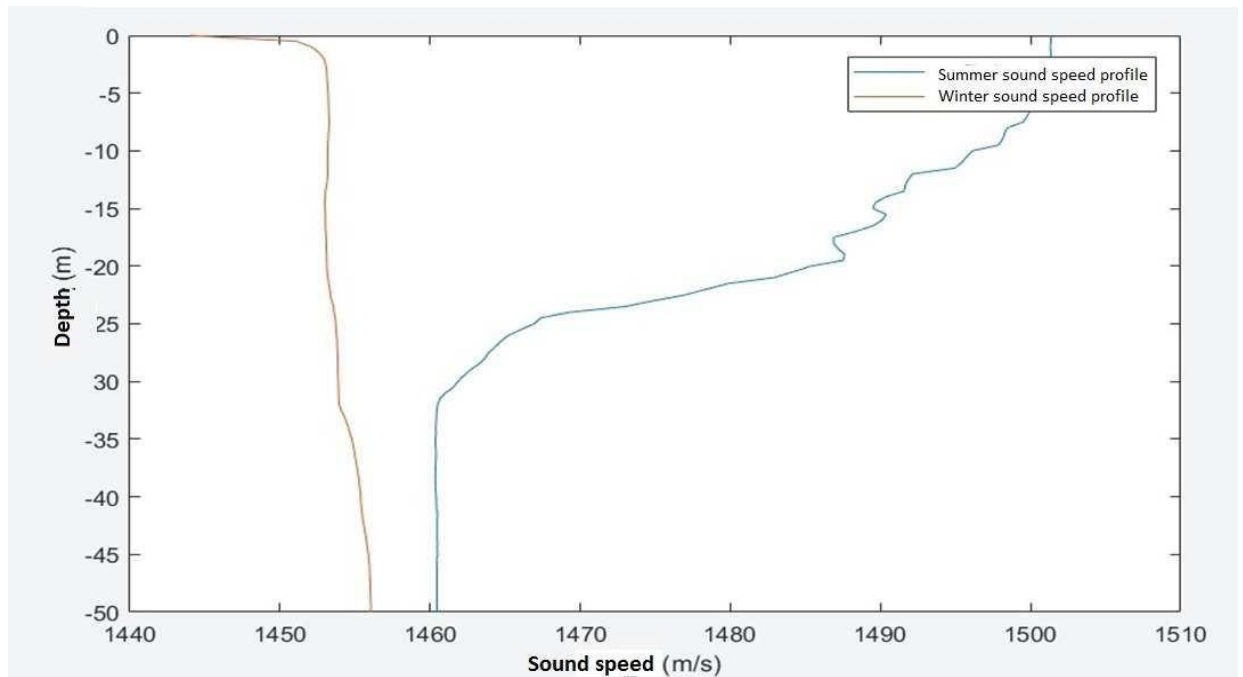


Рисунок 3.6 — Розглянуті профілі швидкості звуку

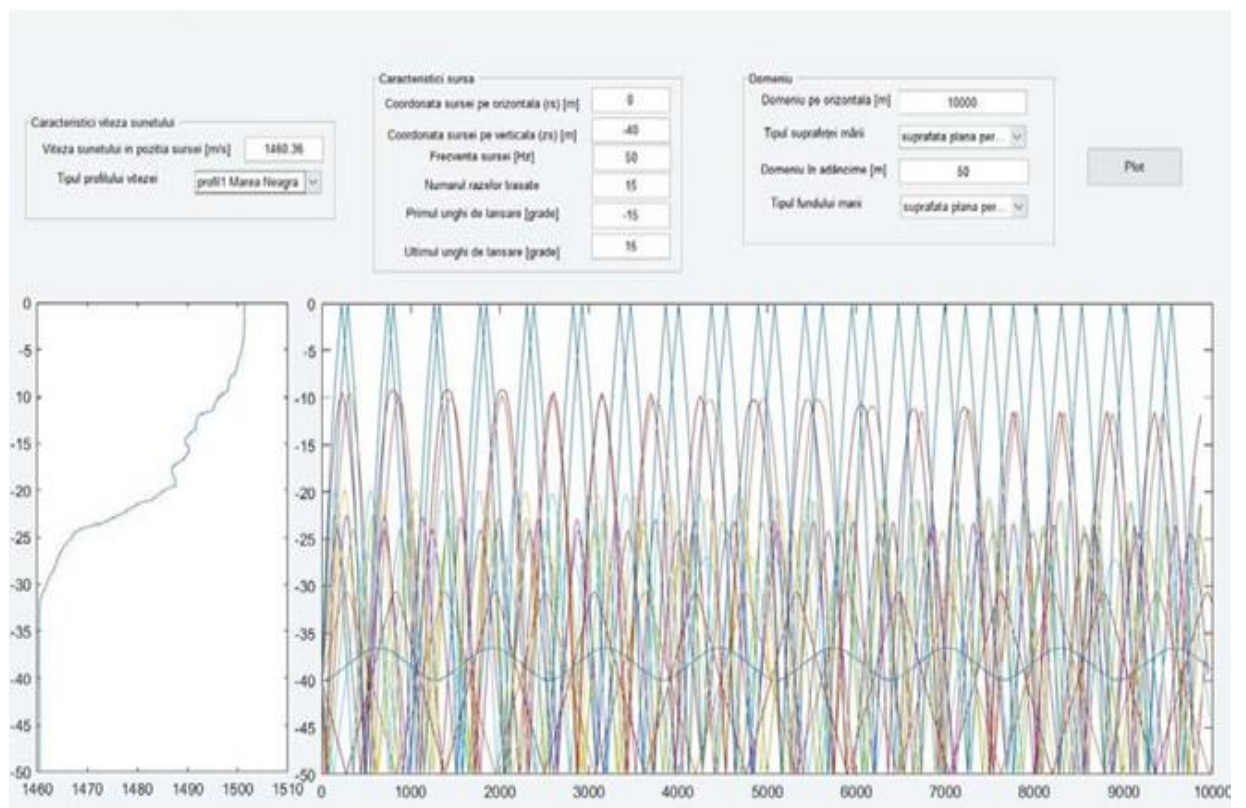


Рисунок 3.6 — Моделювання поширення звукових хвиль — літній профіль, глибина джерела 5м

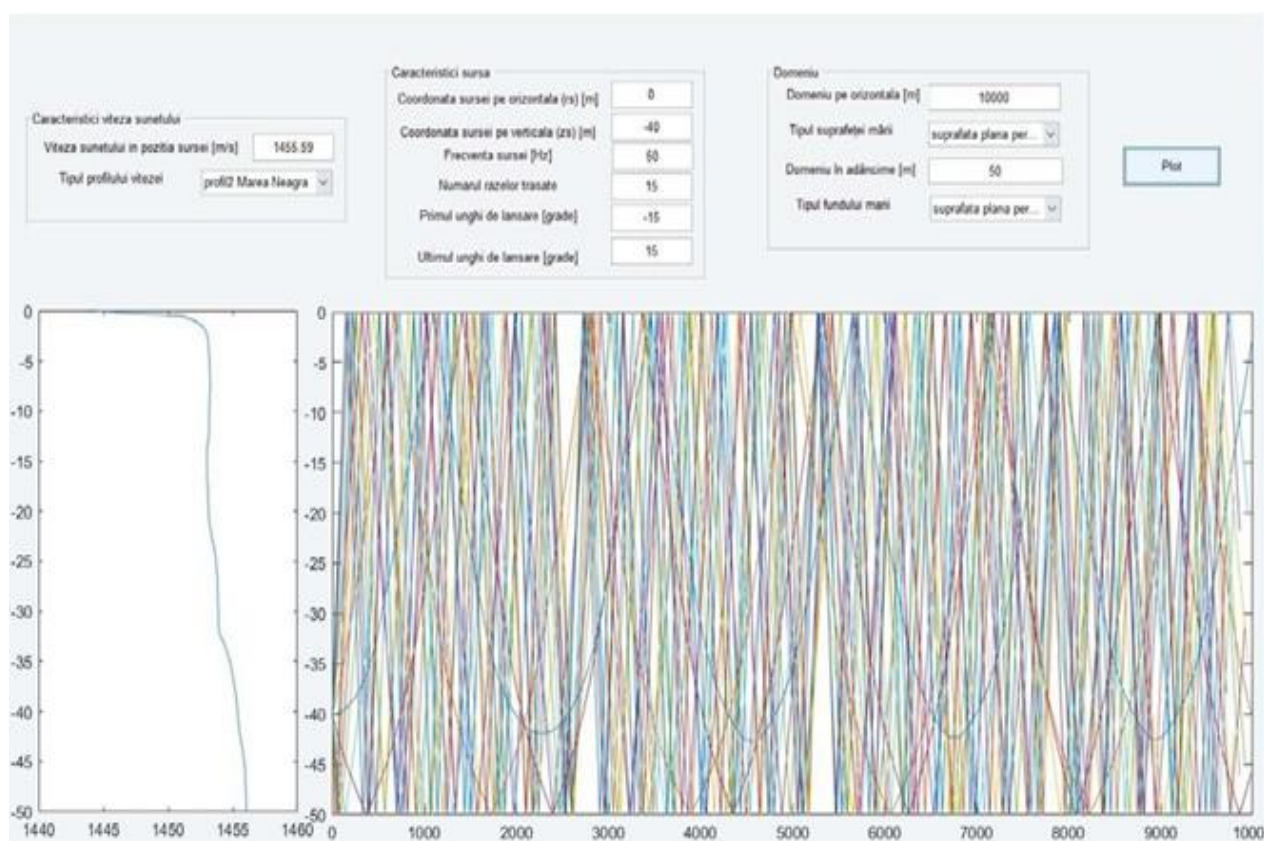


Рисунок 3.7 — Моделювання поширення звукових хвиль — зимній профіль, глибина джерела 40м

Можна спостерігати, що занурення акустичного джерела, підтверджене профілем швидкості звуку, забезпечує цілком повне охоплення досліджуваної ділянки акустичними променями і, отже, мінімізацію площ, де можлива загроза може бути прихована. Цей факт стає більш помітним у глибинах досліджуваної території (глибинах у сотні метрів)[17].

Отже, Розглядаючи морське середовище як акустичний хвилевід, обмежений поверхнею моря і нижче морським дном, поширення акустичних хвиль можна моделювати набором математичних рівнянь щодо ряду параметрів, характерних для морського середовища.

У розробленій імітаційній моделі розглянуто межі - поверхню моря і морське дно — ідеально відбивні плоскі поверхні, і ми не включали втрати поширення, що визначаються частотою сигналу, та пройденою відстанню променя.

Якщо ці обмеження будуть усунені і якщо будуть розглянуті звукові швидкості горизонтального діапазону залежних профілів, то моделювання запропонує кращу апроксимацію поширення звукової хвилі [12].

.

## 4 ЗАСОБИ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

При розробці програмного продукту важливим чинником є правильний вибір технологій та засобів програмної реалізації. Основним середовищем для створення модулю було Microsoft Visual Studio 2017. Для створення інтерфейсу використовувався фреймворк ASP.NET Core. Алгоритми розроблені мовою C# на базі платформи .NET Core. Для робіт з файлами був використаний простір імен System.IO.

### 4.1 Середовище Visual Studio 2017

Середовище розробки Visual Studio(рисунок 4.1) дозволяє швидко і ефективно писати код, не втрачаючи з уваги контекст поточного файлу. Можна легко заглибитися в подробиці, такі як структура виклику, пов'язані функції, повернення і стан тестування. Також доступні рефакторинг коду, знаходження і усунення помилок в коді [10].

Розробники продовжують піклуватися про підвищення продуктивності розробників, створюючи всі умови, щоб сконцентруватися саме на написанні необхідного коду. Наприклад, вже широко використовуються можливості навігації за кодом, рефакторінга, виправлення і налагодженні для всіх підтримуваних мов. Додатково, нові версії дозволяє збільшити швидкість командної розробки з новими real-time функцій модульного тестування і перевірки залежностей.

Новітній, полегшений модульний підхід дозволяє вам встановити тільки ті компоненти середовища, які необхідні і прискорює установку інструменту від початку і до кінця. До того ж, тепер у розробників відпала необхідність створювати проекти і рішення, щоб налагодити будь-який необхідний фрагмент коду.

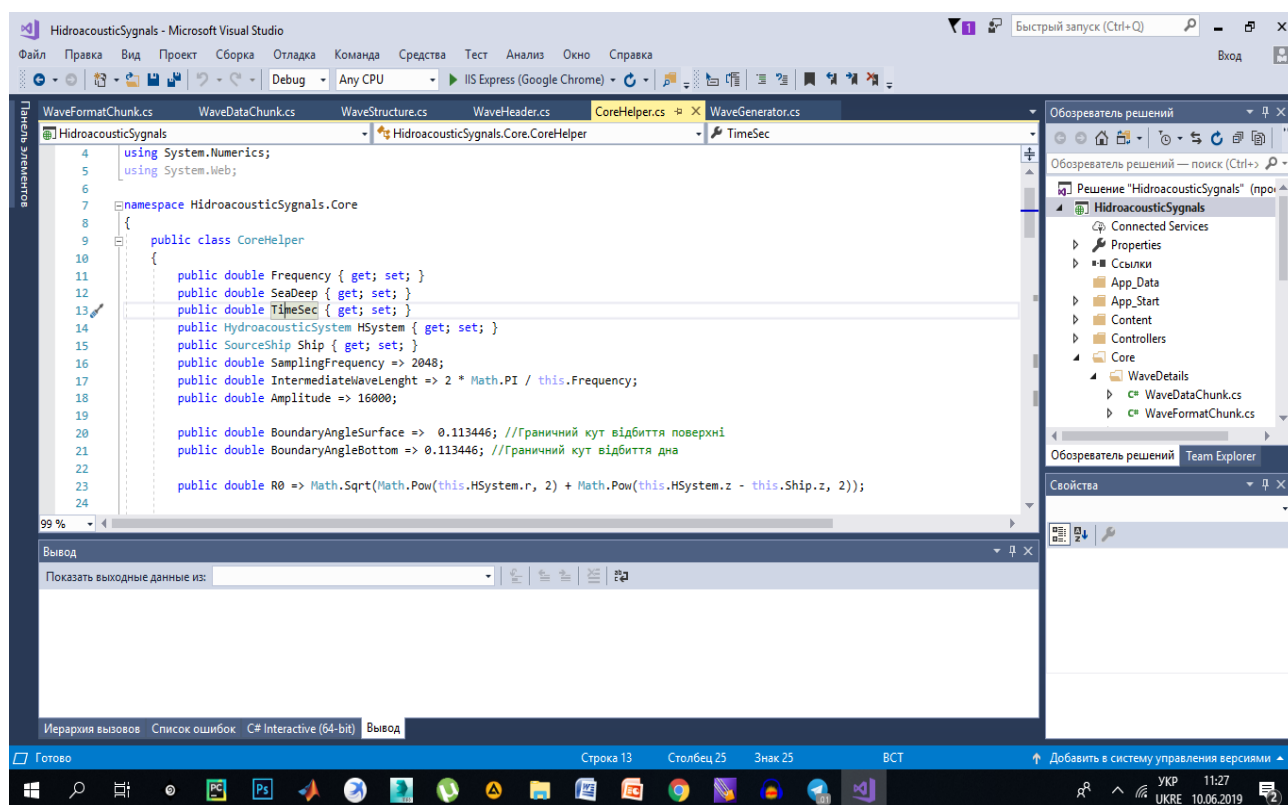


Рисунок 4.1 — Интерфейс середовища Visual Studio

Visual Studio також дозволяє створювати і підключати сторонні додатки (плагіни) для розширення функціональності практично на кожному рівні, включаючи додавання підтримки систем контролю версій вихідного коду (Subversion і VisualSourceSafe), додавання нових наборів інструментів (для редагування і візуального проектування коду на предметно-орієнтованих мовах програмування або інструментів для інших аспектів процесу розробки програмного забезпечення).

Інтуїтивний стиль кодування. За замовчуванням Visual Studio форматує код у міру його введення, автоматично вставляючи необхідні відступи і застосовуючи колірне кодування для виділення елементів типу коментарів. Такі незначні відмінності роблять код більш зручним для читання і менш схильним до помилок. Застосовувані Visual Studio автоматично параметри форматування можна навіть налаштовувати, що дуже зручно у випадках, коли розробник вважає за краще інший

стиль розміщення дужок (наприклад, стиль K & R, при якому відкриває дужка розміщується на тому самому рядку, що і оголошення, якому вона передує).

Більш висока швидкість розробки. Багато з функціональних можливостей Visual Studio спрямовані на те, щоб допомагати розробнику робити свою роботу якомога швидше. Зручні функції, на зразок функції IntelliSense (яка вміє перехоплювати помилки і пропонувати правильні варіанти), функції пошуку і заміни (яка дозволяє відшукувати ключові слова як в одному файлі, так і в усьому проекті) і функції автоматичного додавання і видалення коментарів (яка може тимчасово приховувати блоки коду), дозволяють розробнику працювати швидко і ефективно.

Можливості налагодження. Пропоновані в Visual Studio інструменти налагодження є найкращим засобом для відстеження загадкових помилок і діагностування дивної поведінки. Розробник може виконувати свій код по рядку за раз, встановлювати інтелектуальні точки переривання, при бажанні зберігаючи їх для використання в майбутньому, і в будь-який час переглядати поточну інформацію з пам'яті.

Visual Studio також має і безліч інших функцій: можливість управління проектом; вбудована функція управління вихідним кодом; можливість рефакторизації коду; потужна модель розширюваності. Більш того, в разі використання Visual Studio 2008 Team System розробник отримує розширені можливості для модульного тестування, спільної роботи і управління версіями коду (що значно більше того, що пропонується в більш простих інструментах на кшталт Visual SourceSafe).

## **4.2 Платформа .NET Core**

В наш час .NET Core (рисунки 4.2) — це універсальна платформа розробки з відкритим кодом, яку підтримує корпорація Майкрософт і спільнота .NET на сайті GitHub. .NET Core це модульна реалізація бібліотек і середовищ виконання, в яке



входить піднабір .NET Framework. Вона є кросплатформенною (підтримує Windows, macOS и Linux) і може використовуватися для створення застосунків для приладів, хмар та Інтернету речей. Платформа виконання .NET Core надає:

- систему типів
- функції завантаження збірок
- збирач сміття
- власні функції взаємодії
- інші базові служби.

Програмне забезпечення ASP.NET надає можливості для створення сучасних хмарних додатків, підключених до інтернету: веб-додатків, застосунків інтернету речей, серверної частини мобільних рішень і багато іншого.

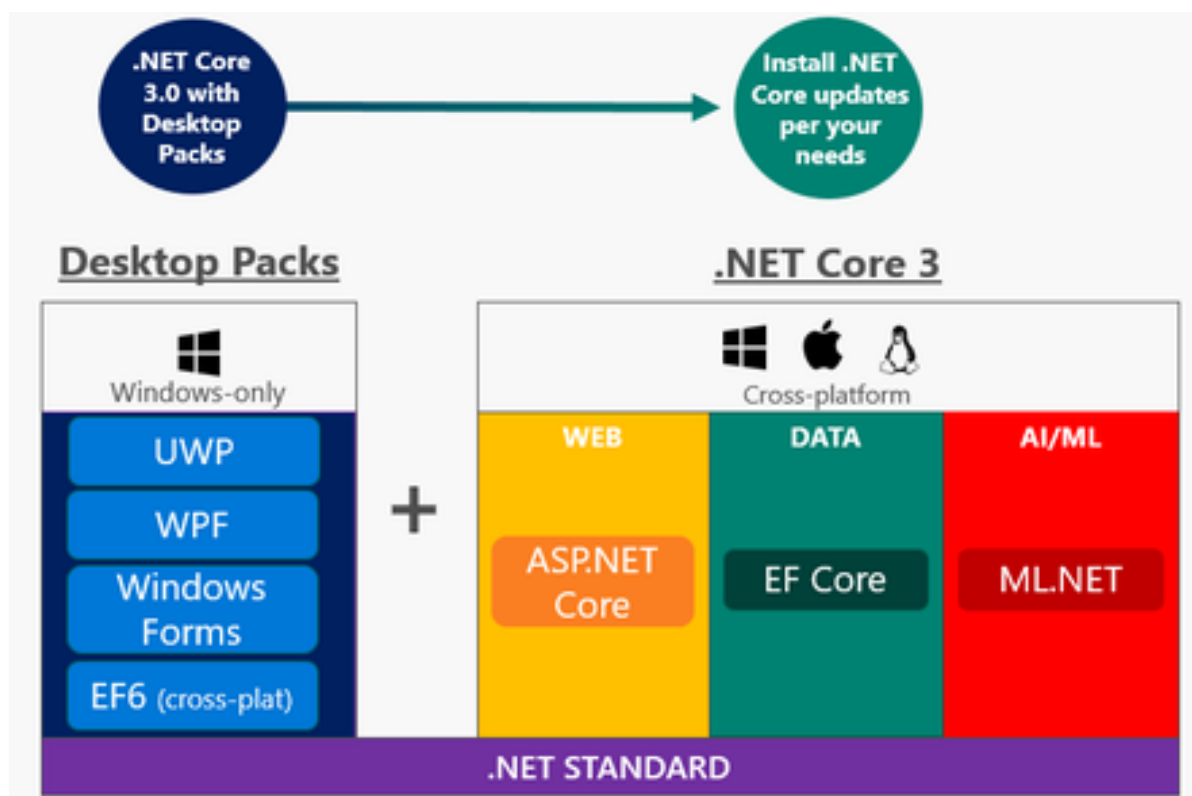


Рисунок — 4.2 .NET Core 3

Засоби .NET Core CLI і компілятори мов (Roslyn и F#) реалізують можливості .NET Core. Використання цієї платформи має багато переваг: ви отримуєте більше

свободи в контролі і зміні проекту, прозорість коду може забезпечити вас інформацією і послужити натхненням для ваших власних проектів на базі .NET Core.

Статус "відкритості" також дає .NET Core велику стійкість, оскільки на відміну від пропрієтарного програмного забезпечення, яке часто буває закинуто творцями, код, що лежить в основі інструментів цієї платформи, завжди буде залишатися загальнодоступним.

У зв'язку з тим, що проект був спроектований відповідно до принципів відкритого ПЗ, платформа .NET Core побудована за допомогою близько 10 000 розробників. Їх внесок включав pull request'и, а також відгуки про все: від дизайну і UX до продуктивності.

Впроваджуючи кращі пропозиції і побажання, команда розробників перетворила .NET Core в платформу, засновану на спільнотах, що робить її більш доступною і ефективною для спільноти розробників, ніж якби вона була створена виключно всередині компанії. Платформа продовжує вдосконалюватися завдяки співпраці, оскільки вона підтримується спільнотою .NET, Microsoft і GitHub. Ви як розробник маєте можливість впливати на майбутнє просування DotNet Core, працюючи з кодом і залишаючи відгуки.

Однією з ключових характеристик є .NET runtime, яка надає .NET Core основні сервіси, включаючи систему типів, збирач сміття, вбудований interop і складання.

Примітивні типи даних, типи композиції додатків і основні утиліти надаються набором бібліотек фреймворка, також відомим як «CoreFX».

Хост додатки «Dotnet» вибирає і запускає виконуючу середу, дозволяючи запускати додатки .NET Core.

У міру розробки ви отримаєте доступ до DotNet Core як пакет розробки програмного забезпечення .NET Core (SDK). Він включає в себе інструменти командного рядка .NET Core, .NET Core і драйвер dotnet - все, що необхідно для створення програми або бібліотеки.

Однією з визначальних особливостей .NET Core є гнучке розгортання: ви можете встановити платформу або як частина свого застосування, або окремо. З FDD ваш пакет розгортання буде менше. Крім того, використання дискового простору і використання пам'яті на пристроях мінімізуються, і ви можете запустити додаток .NET Core з урахуванням будь-якої ОС.

Self-contained розгортання (SCD) включає в себе всі компоненти (також бібліотеки DotNet Core і середу виконання). Цей тип розгортання дає вам повний контроль над версією .NET Core, що використовується вашим додатком. Унікальні характеристики кожного типу розгортання гарантують, що ви зможете розгорнути додаток оптимальним способом в залежності від конкретних потреб.

Ця унікальна програмна платформа вже працює на Windows, Mac OS X і Linux. Хоча це може здатися дивним кроком для Microsoft, але це важливо в технологічному світі, який все більше орієнтується на гнучкість і сегментування, коли справа доходить до операційних систем і платформ.

Доступність .NET Core на інших платформах, відмінних від Windows, робить його кращим кандидатом для використання всіма розробниками, включаючи розробників Mac і Linux, а також дає всієї платформі .NET можливість використовувати зворотний зв'язок. Ця додаткова зворотний зв'язок призводить до продукту, який працює краще для всіх користувачів і дозволяє DotNet Core просувати свої додатки як програмно-які визначаються, а не від платформи визначаються.

Ядро оснащено Agile Development Capability. В рамках крос-сумісності платформа для розробки додатків включає в себе модульну інфраструктуру. Вона видається через NuGet, і ви можете отримати доступ до пакетним функцій, а не до однієї великої збірці. Як розробник ви можете створювати легкі додатки, що містять тільки необхідні пакети NuGet, що зробить вашу програму безпечніше і продуктивніше.

Модульна інфраструктура також дозволяє швидше оновлювати платформу .NET Core, оскільки порушені модулі можуть оновлюватися і випускатися окремо.

Акцент на гнучкість і швидкі релізи поряд з вищезазначеним співпрацею позитивно позиціонує .NET Core в рамках руху DevOps.

Пакет SDK для .NET Core — включає ресурси CLI .NET, середовище виконання ASP.NET Core, а також платформу .NET Core.

Сама платформа .NET Core складається з NuGet-пакетів з виконуваним середовищем, яке встановлює App-Local. Тому додатки можуть носити з собою .NET Core, що дає їм можливість співіснувати з іншими екземплярами .NET Core на одній машині або пристрої. Тому обслуговування виконується індивідуально для кожного додатку і через диспетчер пакетів.

Простір імен System.IO в .NET — це область бібліотек базових класів для служб файлового вводу-виводу, а також вводу-виводу з пам'яті. В даному просторі визначений певний набір класів, інтерфейсів, структур та делегатів. Наприклад клас FileStream дозволяє нам записувати та зчитувати файли.

Для різних форм обрахунків використовується простір імен System.Numerics, що містить структури для роботи з різними видами даних.

## 5 ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ МОДУЛЯ ГЕНЕРАЦІЇ ГІДРОАКУСТИЧНОГО СИГНАЛУ ПРОМЕНЕВИМ МЕТОДОМ

Програмний продукт для генерації гідроакустичного сигналу був написаний за допомогою мови програмування C#, інтерфейс реалізуючий взаємодію з модулем виконаний як веб-інтерфейс за допомогою ASP.NET Core MVC та веб-серверу IIS (Internet Information Services). Дана система є доступною для подальшого використання в більш складних проектах.

Програмний продукт розраховує миттєві значення коливальних швидкостей та сумарного тиску для представлення акустичного поля сигналу у вигляді записаного файлу з даними. Значення розраховуються за допомогою формул променевого метода наведених в третьому пункті.

Для того щоб отримати кінцевий dat-файл з записаним сигналом, спочатку потрібно задати початкові параметри (рисунок 5.1):

Після введення параметрів у відповідні поля потрібно натиснути генерувати сигнал (рисунок 5.1), після чого програмний продукт обрахує та запише сигнал в директорію, вказану в програмному коді, як audio.dat та запропонує зберегти копію файлу у вказану користувачем директорію (рисунок 5.2). У файл спочатку записується сформований заголовок, а потім семпли сигналу з четвірок даних  $V_x$ ,  $V_y$ ,  $V_z$ ,  $P$ , виміри проводиться по часу моделювання з кроком кожному  $\frac{1}{\text{Частота дискретизації}}$  частину секунди, тобто частота дискретизації це кількість вимірів на секунду.  $V_x$ ,  $V_y$ ,  $V_z$  — значення коливальної швидкості в даний момент часу,  $P$  — миттєве значення сумарного тиску. Всі числа приводяться до типу short(шістнадцяткове зі знаком). Вихідний файл має структуру описану в таблиці 5.1. Два нулі це заголовок файлу.

### Hidroacoustic sygnals

Координата джерела сигналу по(z):

Координати ГАС(x,y,z):

Частота випромінювання джерела:

Час моделювання сигналу:

Глибина моря:

Згенерувати сигнал

Рисунок 5.1 — Веб-інтерфейс для роботи з модулем

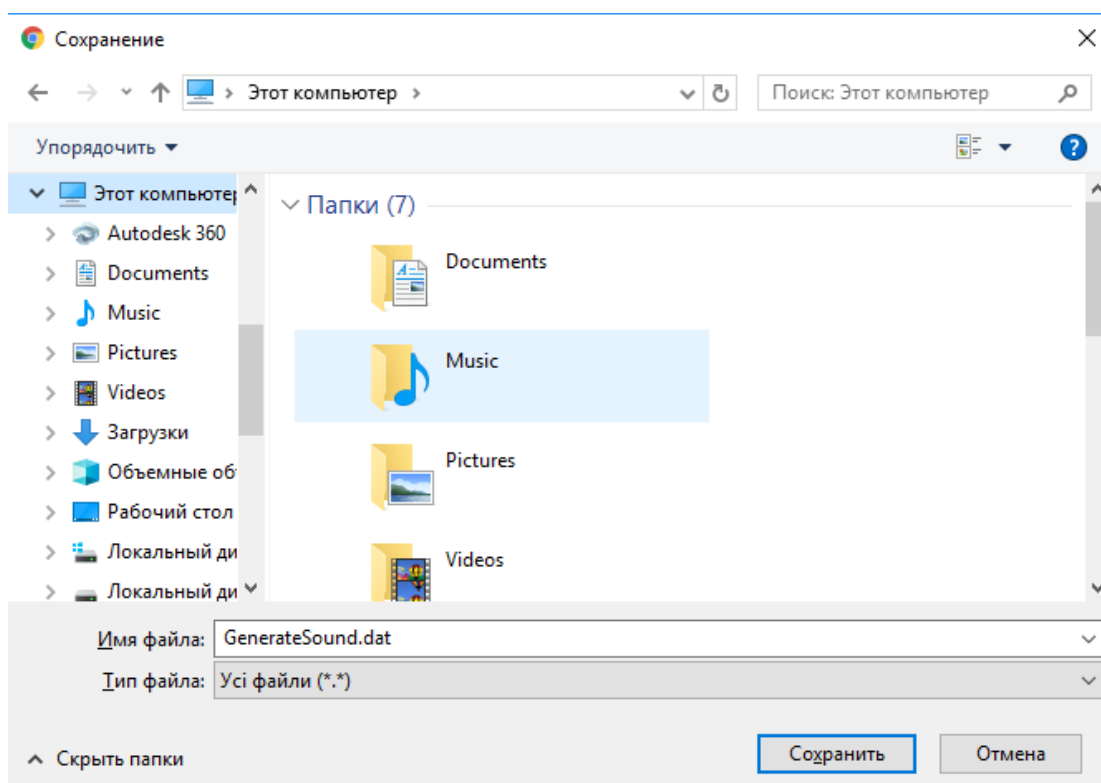


Рисунок 5.2 — Збереження копії змодельованого сигналу

Таблиця 5.1 — структура запису вихідного файлу

0	0		
X1	Y2	Z1	P1
X2	Y2	Z1	P1
• • •			
Xn	Yn	Zn	Pn

Отриманий гідроакустичний сигнал має задану частоту дискретизації та є нормованим по амплітуді, ці параметри можуть бути легко змінені в коді програмного модулю. Для перевірки сигнал може бути відкритий за допомогою аудіо редактору звукових файлів — Audacity. Імпортувати файл потрібно, як Raw-дані з кодуванням Signed 16-bit PCM, байтів порядок — Big-endian, з 4 каналами та частотою дискретизації певною частотою дискретизації за замовчуванням це 2048 Гц (рисунок 5.3).

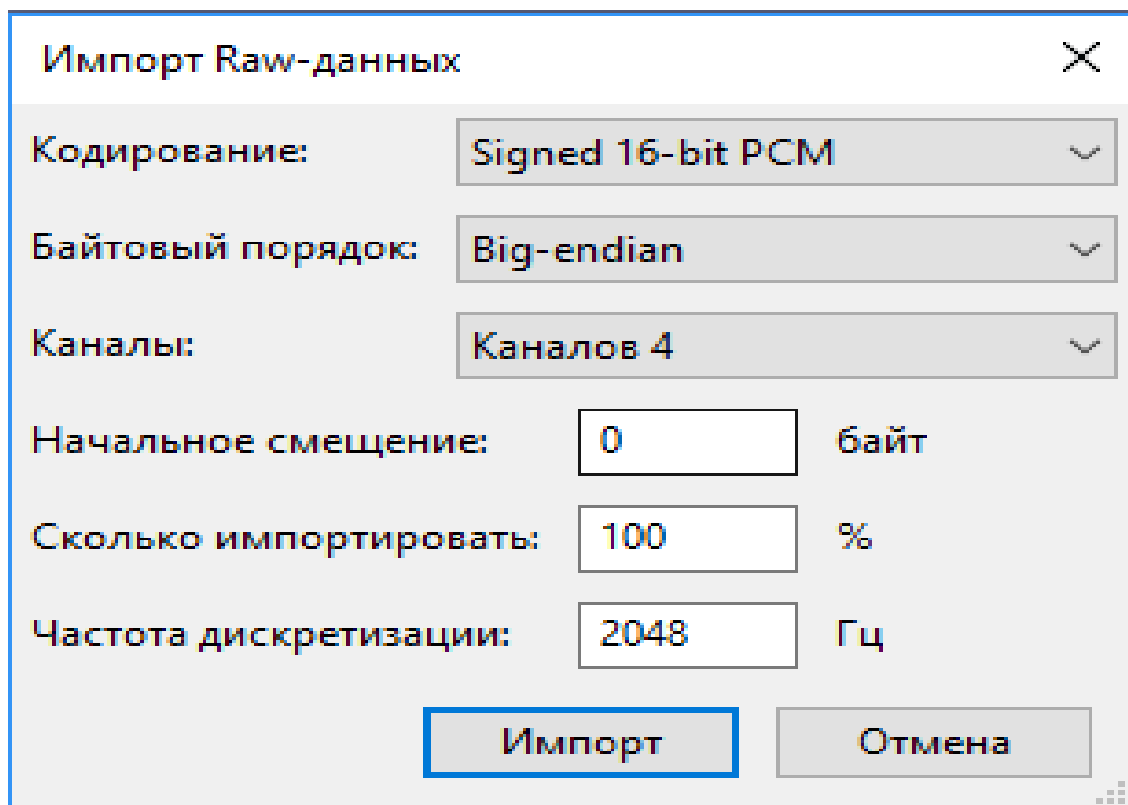


Рисунок 5.3 — Імпорт даних до Audacity

Після імпорту ми маємо відображення сигналу у вигляді гармонічних коливань компонентів. Даний аудіо редактор надає змогу переглянути та оцінити сигнал по різним характеристикам, для перевірки даних оцінимо графік спектру (рисунок 5.5).

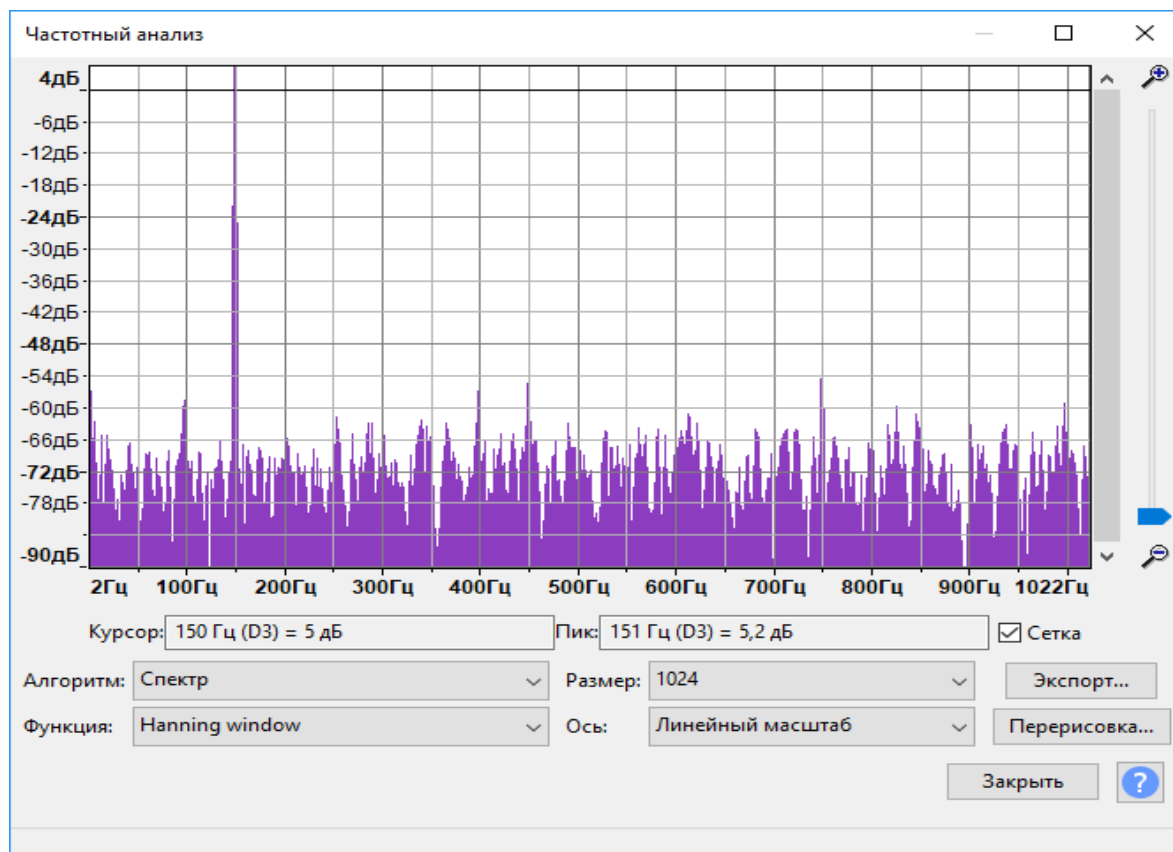


Рисунок 5.4 — Частотний спектр змодельованого сигналу з параметрами Рисунок 5.1

Наглядно принцип роботи програмного продукту можна уявити за допомогою діаграми класів(рисунок 5.5).

Програма отримує на вхід дані про гідроакустичні об'єкти, амплітуду і глибину хвильоводу. Клас Controller методами класу ініціалізує об'єкти класів CoreHelper (та внутрішніх класів), WaveModeller,. WaveStructure містить метод Save(), що зберігає вихідний файл, а отримує інформацію про структуру запису з класів WaveFormatChunk, WaveDataChunk та WaveHeader.



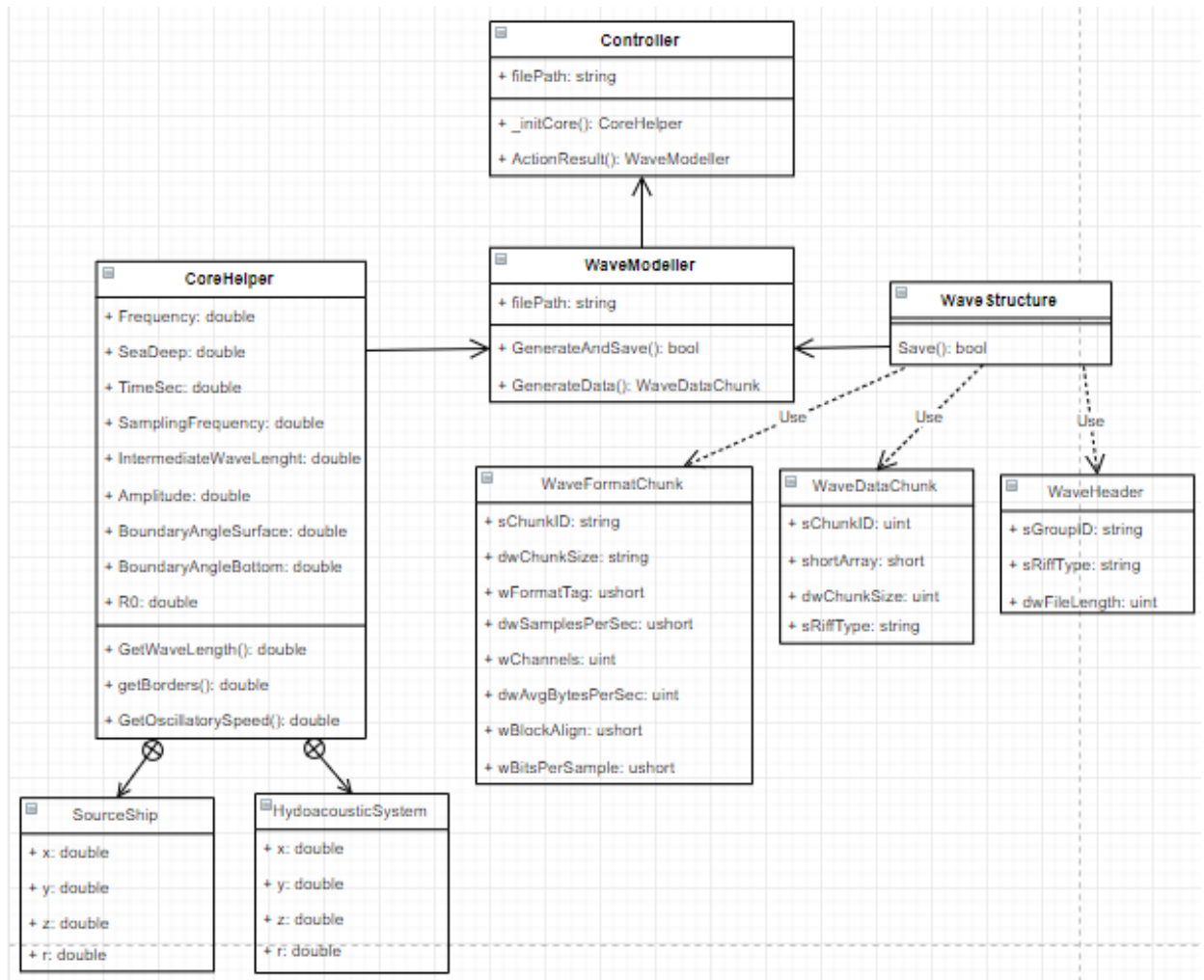


Рисунок 5.5 — Діаграма класів програмного модулю

WaveModeller використовує методи класів CoreHelper і WaveStructure, за допомогою яких обраховує та записує дані в файл. CoreHelper містить основні параметри за якими моделюється сигнал. Класи SourceShip і HydroacousticSystem містять інформацію про джерело та приймаючу сигнал систему відповідно.

## ВИСНОВКИ

У ході виконання дипломної роботи було створено додаток, що містить функціонал для моделювання гідроакустичних сигналів в межах плоско-паралельного хвилеводу.

Програмний продукт написано на мові C# та на платформах .NET Core

Створений функціонал дозволяє виконувати задачі

— моделювання гідроакустичного сигналу;

— запис змодельованих даних до файлу.

Програмний продукт є універсальним і може бути використаний для більш масштабного проекту, який спеціалізується на моделюванні гідроакустичних об'єктів.

Користувач має змогу власноруч задавати всі параметри вхідних даних. На виході генерується єдиний сигнал, який і є результатом виконання генерації, та представлений у вигляді dat-файлу.

Користувач має змогу зберегти dat-файл, змінюючи його ім'я та вибравши директорію для збереження.

Дана система істотно спрощує процес моделювання гідроакустичних сигналів та дозволяє досягти чіткого результату в межах більш складних систем.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Доклады XII научной школы-семинара им. акад. Л.М.Бреховских "Акустика океана", М.: ГЕОС, 2009, 486 с.
2. Hovem, J. M. (2010). Marine Acoustics: The Physics of Sound in Underwater Environments. Peninsula Publishing, Los Altos, California, USA.
3. Francois, R. E., and G. R. Garrison. 1982. Sound absorption based on ocean measurements, 72(6), 1879–1890.
4. Распространение волн и подводная акустика / Под ред. Дж.Б. Келлера и Дж. Пападакиса. – М.: Мир, 1980. – 230 с.
5. Алексеев Г.В., Комашинская Т.С. Об активной минимизации потенциальной энергии звукового поля в двумерном многомодовом волноводе // Акустический. журнал. – 2003. – Т. 49, №2. – С. 149–155.
6. Boumenir A. Sampling and eigenvalues of non-self-adjoint Sturm-Liouville problems // SIAM J. Sci. Comput. – 2001. – Vol. 23, N 1. – P. 219–229.
7. Abrosimov. Подласов Е.С. О моделировании волновых процессов в рамках лучевого приближения // Компьютерная математика. – 2005.
8. Lee D., McDaniel S.T. Ocean acoustic propagation by finite difference
9. Келлер Дж. П. Распространение волн и подводная акустика / Пападакис Дж. Келлер Дж., 1980. – 280 с
10. Microsoft Visual Studio [Электронный ресурс] – Режим доступа до ресурсу: <https://www.visualstudio.com/>.
11. Rodriguez O.C. Traceo Ray tracing program [Электронный ресурс] / Rodriguez O.C.. – 2011. – Режим доступа до ресурсу: [www.siplab.fct.ualg.pt](http://www.siplab.fct.ualg.pt).
12. Modeling and simulation of hydro-acoustic signal propagation in the Romanian Black Sea coast / E.G. Curcă, R.G. Damian, O. Radu, V. Roșca. – Research Center for Navy, Țefănită Vodă Street, N, 2015.
13. Arctan - Powerful Acoustic Simulation Software [Электронный ресурс] – Режим доступа до ресурсу: <https://www.mscsoftware.com/>.

14. Fox D N. The modular Ocean Data Assimilation System Oceanography / Fox D N., 2002. – (1). – (15).
15. Leroy C C. A new equation for the accurate calculation of sound speed in all oceans / Leroy C C. – .Acoust. Soc. Am, 2008. – 124 с
16. Jensen F B. Computational Ocean Acoustics / Jensen F B., 2011. – (2)
17. Авилов К.В., Добряков Н.А., Попов О.Е. Комплекс программных средств для вычисления звуковых полей в морской среде, неоднородной по глубине и трассе распространения // Доклады X научной школы-семинара академика Л. М. Бреховских "Акустика океана", совмещенной с XIV сессией Российского акустического общества. – М.: ГЕОС, 2004. – С. 27–30.

## ДОДАТОК 1

Модуль генерації гідроакустичного сигналу в плоско-паралельному хвилеводі  
променевим методом

Специфікація

УКР.НТУУ"КПІ ім. Ігоря Сікорського"\_ТЕФ\_АПЕПС\_ТР5155\_19Б

Аркушів 2

Київ – 2019

Позначення	Найменування	Примітки
Документація		
УКР.НТУУ" КПІ ім. Ігоря Сікорського" _ТЕФ_АПЕПС_ТР5 155_19Б	Записка.doc	Пояснювальна записка
Компоненти		
УКР.НТУУ" КПІ ім. Ігоря Сікорського" _ТЕФ_АПЕПС_ТР5 155_19Б 12-1	_ViewStart.cshtml	Веб-інтерфейс системи
УКР.НТУУ" КПІ ім. Ігоря Сікорського" _ТЕФ_АПЕПС_ТР5 155_19Б 12-2	WaveGenerator.cs	Модуль програми для обрахунку та запису сигналу
УКР.НТУУ" КПІ ім. Ігоря Сікорського" _ТЕФ_АПЕПС_ТР5 155_19Б 12-3	CoreHelper.cs	Модуль для класів, що представляють джерело, ГАС та сигнал

## ДОДАТОК 2

Модуль для розрахунку сумарного акустичного тиску та компонентів  
коливальної швидкості

Лістинг програмного модулю

УКР.НТУУ"КПІ ім. Ігоря Сікорського"\_ТЕФ\_АПЕПС ТВ51148\_19Б 12-2

Аркушів 7

Київ – 2019

```

using System;
using System.Collections.Generic;
using System.Collections.Specialized;
using System.IO;
using System.Linq;
using System.Numerics;
using System.Web;
using System.Windows;
using HidroacousticSignals.Core.WaveDetails;

namespace HidroacousticSignals.Core
{
    public class WaveModeller
    {
        public const string filePath = @"C:\Users\User\Desktop\audio.dat"; //директорія для запису сигналу

        public WaveStructure wave { get; set; }
        public CoreHelper core { get; set; }

        public WaveModeller(CoreHelper _core)
        {
            this.core = _core;
        }

        public bool GenerateAndSave()
        {
            var header = new WaveHeader();
            var format = new WaveFormatChunk(this.core);
            var data = this.GenerateData(format);

            var waveHelper = new WaveStructure(header, format, data);
            return waveHelper.Save(filePath);
        }
    }
}

```



```

private WaveDataChunk GenerateData(WaveFormatChunk format)
{
    this.core.getBorders(out double bottom, out double top);
    var core = this.core;
    uint numSamples = (uint) (this.core.SamplingFrequency * 4 * this.core.TimeSec * 60);
    var dataChunk = new WaveDataChunk(numSamples);
    double k = (core.Frequency * (2 * Math.PI)) / 1450;
    int g = 0;

    try
    {

        for (double i = 0; i < numSamples; i++ , g +=4) //основний цикл по часу моделювання
        {
            double t = i / this.core.SamplingFrequency;
            double p = 0;
            double pRez = 0;
            var resVector = new Vector2(0, 0);
            double normAmplitude = (this.core.Amplitude / (top + bottom));

            for (var n = -1*bottom ; n <= top; n++) //цикл для розрахунку миттєвих значень компонентів по
кількості
            {
                // променів

                double waveLength = core.GetWaveLength(n);

                double r2 = Math.Pow(core.HSystem.r, 2);

                double RI = Math.Sqrt(r2 + waveLength); //Довжина n-го променю

                double partOfFactor = RI / this.core.IntermediateWaveLenght;

```

```
double modfi = partOfFactor - Math.Truncate(partOfFactor);
double deltaFi = (modfi)*2*Math.PI; //Знаходимо дельта fi
```

```
float x = (float)(normAmplitude * Math.Cos(deltaFi));
float y = (float)(normAmplitude * Math.Sin(deltaFi));
//float z = (float)(normAmplitude * Math.Sqrt(Math.Pow(x,2) + Math.Pow(y,2))/core.R0);
//vectorPD = new Vector3(x, y, 0);
resVector += new Vector2(x, y);
```

```
double positiveWaveL = Math.Sqrt(r2 + core.GetWaveLength(n, 1));
double negativeWaveL = Math.Sqrt(r2 + core.GetWaveLength(n, -1));
double firstPart = Math.Pow(-1 , n);
```

```
p += normAmplitude * firstPart * ((Math.Cos(k * negativeWaveL ) / negativeWaveL)
    - (Math.Cos(k * positiveWaveL ) / positiveWaveL)); // розрахунок сумарного тиску
```

```
}
```

```
var res = new Vector2(0, 0);
```

```
res = resVector;
```

```
pRez = p * Math.Cos(t *this.core.Frequency * 2 * Math.PI) * (0.25 * Math.PI);
```

```
double Vx = core.GetOscillatorySpeed(t, resVector, Math.Sqrt( Math.Pow(res.X, 2) + Math.Pow(res.Y,
2))/core.R0);
```

```
double Vy = core.GetOscillatorySpeed(t, resVector, Math.Sqrt(Math.Pow(res.Y, 2)) /
Math.Sqrt(Math.Pow(res.X, 2) + Math.Pow(res.Y, 2)));
```

```
double Vz = core.GetOscillatorySpeed(t, resVector, Math.Sqrt(Math.Pow(res.X, 2) + Math.Pow(res.Y,
2)) / core.SeaDeep); //отримання значень коливальної швидкості для трьох проекцій
```

```
//запис значень до масиву
```

```
dataChunk.shortArray[g] = (short)Convert.ToInt16(Vx);
```

```
dataChunk.shortArray[g +1] = (short)Convert.ToInt16(Vy);
```

```
dataChunk.shortArray[g +2] = (short)Convert.ToInt16(Vz);
```

```

        dataChunk.shortArray[g + 3] = (short)Convert.ToInt16(pRez);
    }

}

catch (Exception ex) { }

// Підрахунок розміру блоку даних в байтах
dataChunk.dwChunkSize = (uint)(dataChunk.shortArray.Length * (format.wBitsPerSample / 8));
return dataChunk;
}

}

using System;
using System.Collections.Generic;
using System.Linq;
using System.Numerics;
using System.Web;

namespace HidroacousticSignals.Core
{
    public class CoreHelper
    {
        public double Frequency { get; set; }
        public double SeaDeep { get; set; }
        public double TimeSec { get; set; }
        public HydroacousticSystem HSystem { get; set; }
        public SourceShip Ship { get; set; }
        public double SamplingFrequency => 2048;
        public double IntermediateWaveLenght => 2 * Math.PI / this.Frequency;
        public double Amplitude => 16000;

        public double BoundaryAngleSurface => 0.113446; //Граничний кут відбиття поверхні
    }
}

```

```
public double BoundaryAngleBottom => 0.113446; //Граничний кут відбиття дна
```

```
public double R0 => Math.Sqrt(Math.Pow(this.HSystem.r, 2) + Math.Pow(this.HSystem.z - this.Ship.z, 2));
```

```
public CoreHelper(HydroacousticSystem system, SourceShip ship, double freq, double deep, double time)
{
    this.HSystem = system;
    this.Ship = ship;
    this.Frequency = freq;
    this.SeaDeep = deep;
    this.TimeSec = time ;
}
```

```
public double GetWaveLength(double n, int? hardSign = null)
{
    double insidePow;

    if (hardSign != null)
    {
        insidePow = (this.HSystem.z + (hardSign.Value) * this.Ship.z - 2 * n * this.SeaDeep);
    }
    else
    {
        insidePow =(this.HSystem.z + (n <= 0 ? 1 : -1) * this.Ship.z - 2 * n * this.SeaDeep);
    }

    return Math.Pow(insidePow, 2);
}

public void getBorders(out double surfaceBorder, out double bottomBorder)
{

```

```

        double lenOfFirstPartRepulseSurface = Math.Sqrt(Math.Pow(this.Ship.z, 2)) *
Math.Tan(this.BoundaryAngleSurface);

        double lenOfOneFullRepulseSurface = this.SeaDeep * Math.Tan(this.BoundaryAngleSurface);

        double fullCountOfRepulseSurface = (this.HSystem.r - lenOfFirstPartRepulseSurface) /
lenOfFirstPartRepulseSurface;


        double lenOfFirsPartRepulseBottom =(this.SeaDeep - Math.Sqrt(Math.Pow(this.Ship.z, 2))) *
Math.Tan(this.BoundaryAngleBottom);

        double lenOfOneFullRepulseBottom = this.SeaDeep * Math.Tan(this.BoundaryAngleBottom);

        double fullCountOfRepulseBottom = (this.HSystem.r - lenOfFirsPartRepulseBottom) /
lenOfOneFullRepulseBottom;


        surfaceBorder = (int)fullCountOfRepulseSurface + 1; //розрахунок кількості променів, що відбилися від
поверхні

        bottomBorder = (int)fullCountOfRepulseBottom + 1; // розрахунок кількості променів, що відбилися
від дна


public double GetOscillatorySpeed(double time, Vector2 sumVector, double param)
{
    var arctgFi = Math.Atan2(sumVector.Y, sumVector.X)*(180/Math.PI);
    double anglePart = (this.Frequency * 2 *Math.PI * time)+arctgFi;

    return Math.Sin(anglePart) * (param)/R0;//розрахунок коливальних швидкостей
}


public class HydroacousticSystem
{
    public double x { get; set; }
    public double y { get; set; }
    public double z { get; set; }
    public double r { get; set; }

    public HydroacousticSystem(double x, double y, double z)
    {

```

```

        this.x = x;

        this.y = y;

        this.z = z;

        this.r = Math.Sqrt(Math.Pow(x,2)+ Math.Pow(y,2));
    }
}

```

public class SourceShip//клас що реалізує абстрактне представлення джерела сигналу

```

{
    public double x { get; set; }
    public double y { get; set; }
    public double z { get; set; }

```

```

    public double r { get; set; }

```

public SourceShip(double z) //клас що реалізує абстрактне представлення джерела сигналу

```

{
    this.x = 0;

    this.y = 0;

    this.z = z;

```

```

    this.r = 0;

```

```

}

```

```

}

```

```

}

```

```

}

```

## ДОДАТОК 3

Модуль генерації гідроакустичного сигналу в плоско-паралельному хвилеводі  
променевим методом

Опис програмного модулю

УКР.НТУУ"КПІ ім. Ігоря Сікорського"\_ТЕФ\_АПЕПС ТВ51148\_19Б 12-2

Аркушів 8

Київ – 2019

## АНОТАЦІЯ

Додаток надає лаконічний та структурний опис реалізації модулю для генерації гідроакустичного сигналу променевим методом.

Програмне забезпечення надає можливість моделювання гідроакустичного сигналу променевим методом в плоско-паралельному хвилеводі при заданих параметрах акваторію.

Програмне забезпечення виконане за допомогою мови програмування C# у середовищі VisualStudio 2017 у вигляді веб-сервісу за технологією ASP.NET MVC.



## ЗМІСТ

1. Загальні відомості.....	4
2. Функціональне призначення .....	5
3. Опис логічної структури .....	6
4. Технічні засоби, що використовувалися.....	7
5. Вхідні та вихідні дані .....	8

## ЗАГАЛЬНІ ВІДОМОСТІ

Програмний код модулю міститься в межах класу WaveModeller, що включає в себе методи для розрахунку сумарного акустичного тиску та компонентів коливальної швидкості та запису їх в файл даних.

Модуль є ядром програмного продукту, адже виконує операції для моделювання сигналу.

Необхідні дані про акваторій та сигнал сервіс приймає з класу Controller, який в свою чергу отримує їх з веб-сервісу за технологією ASP.NET MVC.

-5-

## **ФУНКЦІОНАЛЬНЕ ПРИЗНАЧЕННЯ**

Програмний продукт реалізує такий функціонал:

- веб-інтерфейс для отримання вхідних даних;
- моделювання гідроакустичного сигналу променевим методом в плоско-паралельному хвилеводі;
- запис отриманих даних до .dat-файлу;

-6-

## **ОПИС ЛОГІЧНОЇ СТРУКТУРИ**

Моделювання сигналу реалізовано у формі класу, який моделює сигнал методом уявних джерел з заданими параметрами та має наступні методи:

- 1) `GenerateData ()`;
- 2) `GenerateAndSave ()`;

-7-

## **ТЕХНІЧНІ ЗАСОБИ, ЩО ВИКОРИСТОВУВАЛИСЯ**

Програмний код модулю був виконаний за допомогою мови програмування C# у середовищі VisualStudio 2017 на базі платформи NET Core.

Для реалізації отримання вхідних даних розроблено веб-інтерфейс за технологією ASP.NET MVC.

## **ВХІДНІ ТА ВИХІДНІ ДАНІ**

Вхідними даними є:

- Координати джерела сигналу;
- Координати Гідроакустичної системи(ГАС);
- Частоту випромінювання джерела сигналу;
- Час моделювання;
- Глибина моря.

Вихідними даними є:

- записаний у dat-файл змодельований сигнал;